

STMicroelectronics

Trusted Platform Module ST33KTPM2XSPI / ST33KTPM2X / ST33KTPM2A / ST33KTPM2I

FIPS 140-3 Non-Proprietary Security Policy

Document Version: 01-00
Date: 2026-09-09

Public Material – May be reproduced only in its original entirety (without revision).

Table of Contents

1.1 Overview 6
1.2 Security Levels 6
2 Cryptographic Module Specification 7
2.1 Description 7
2.2 Tested and Vendor Affirmed Module Version and Identification 9
2.3 Excluded Components 13
2.4 Modes of Operation 13
2.5 Algorithms 14
2.6 Security Function Implementations 18
2.7 Algorithm Specific Information 28
2.8 RBG and Entropy 29
2.9 Key Generation 29
2.10 Key Establishment 30
2.11 Industry Protocols 30
3 Cryptographic Module Interfaces 31
3.1 Ports and Interfaces 31
3.2 Pinout description 32
4 Roles, Services, and Authentication 35
4.1 Authentication Methods 35
4.2 Roles 35
4.3 Approved Services 35
4.4 Non-Approved Services 83
4.5 External Software/Firmware Loaded 88
5 Software/Firmware Security 89
5.1 Integrity Techniques 89
5.2 Initiate on Demand 89
6 Operational Environment 90
6.1 Operational Environment Type and Requirements 90
7 Physical Security 91
7.1 Mechanisms and Actions Required 91
7.2 User Placed Tamper Seals 91
7.3 Filler Panels 91
7.4 Fault Induction Mitigation 91

Public Material – May be reproduced only in its original entirety (without revision).

7.5 EFP/EFT Information 92
7.6 Hardness Testing Temperature Ranges 92
8 Non-Invasive Security 93
8.1 Mitigation Techniques 93
9 Sensitive Security Parameters Management 94
9.1 Storage Areas 94
9.2 SSP Input-Output Methods 94
9.3 SSP Zeroization Methods 96
9.4 SSPs 96
9.5 Transitions 103
9.6 Additional Information 103
10 Self-Tests 104
10.1 Pre-Operational Self-Tests 104
10.2 Conditional Self-Tests 104
10.3 Periodic Self-Test Information 106
10.4 Error States 106
11 Life-Cycle Assurance 108
11.1 Installation, Initialization, and Startup Procedures 108
11.2 Administrator Guidance 108
11.3 Non-Administrator Guidance 108
11.4 Design and Rules 108
11.5 Maintenance Requirements 109
11.6 End of Life 109
12 Mitigation of Other Attacks 110
References and Definitions 111

Public Material – May be reproduced only in its original entirety (without revision).

List of Tables

Table 1: Security Levels 6
Table 2: Tested Module Identification – Hardware 9
Table 3 – KE2 Module Configuration with FW 9.257 11
Table 4 – KE2 Module Configuration with FW 9.258 11
Table 5 – KE3 Module Configuration with FW 9.257 11
Table 6 – KG8 Module Configuration 12
Table 7 – KG8 Module Configuration with FW 9.258 12
Table 8 – KG9 Module Configuration 12
Table 9 – KJ5 Module Configuration 13
Table 10 – ZA9 Module Configuration 13
Table 11 – AC5 Module Configuration 13
Table 12: Modes List and Description 14
Table 13: Approved Algorithms 16
Table 14: Vendor-Affirmed Algorithms 17
Table 15: Non-Approved, Allowed Algorithms with No Security Claimed 17
Table 16: Non-Approved, Not Allowed Algorithms 18
Table 17: Security Function Implementations 28
Table 18: Entropy Certificates 29
Table 19: Entropy Sources 29
Table 20: Ports and Interfaces 31
Table 21 – Pins Definition 33
Table 22: Roles 35
Table 23 – Mapping between services 36
Table 24: Approved Services 82
Table 25: Non-Approved Services 87
Table 26: Mechanisms and Actions Required 91
Table 27: EFP/EFT Information 92
Table 28: Hardness Testing Temperatures 92
Table 29: Storage Areas 94
Table 30: SSP Input-Output Methods 95
Table 31: SSP Zeroization Methods 96
Table 32: SSP Table 1 98
Table 33: SSP Table 2 102
Table 34 – Security Strength of a Key Depending on the Underlying Algorithm Used and its Size 103
Table 35: Pre-Operational Self-Tests 104
Table 36: Conditional Self-Tests 105
Table 37: Pre-Operational Periodic Information 106
Table 38: Conditional Periodic Information 106
Table 39: Error States 107
Table 40 – References 112
Table 41 – Acronyms and Definitions 113

Public Material – May be reproduced only in its original entirety (without revision).

List of Figures

Figure 1 – HW block diagram 8
Figure 2 – UFQFPN32 Package 10
Figure 3 – UFQFPN32 WF Package 10
Figure 4 – TSSOP20 Package 10
Figure 5 – WLCSP24 Package 10
Figure 6 – UFQFPN32 / UFQFPN32 WF Pinout Diagram 33
Figure 7 – TSSOP20 Pinout Diagram 34
Figure 8 - WLCSP24 Pinout Diagram 34
Figure 9 – Firmware block diagram 89

Public Material – May be reproduced only in its original entirety (without revision).

1 General

1.1 Overview

This document is the non-proprietary FIPS 140-3 Security Policy for the STMicroelectronics Trusted Platform Module ST33KTPM2X / ST33KTPM2XSPI / ST33KTPM2A / ST33KTPM2I. It contains the security rules under which the module must operate and describes how this module meets the requirements as specified in FIPS PUB 140-3 (Federal Information Processing Standards Publication 140-3) for an overall Security Level 1 module.

1.2 Security Levels

The FIPS 140-3 security levels for the module are listed in table below:

Section	Title	Security Level
1	General	1
2	Cryptographic module specification	1
3	Cryptographic module interfaces	1
4	Roles, services, and authentication	1
5	Software/Firmware security	1
6	Operational environment	1
7	Physical security	3
8	Non-invasive security	N/A
9	Sensitive security parameter management	1
10	Self-tests	1
11	Life-cycle assurance	1
12	Mitigation of other attacks	N/A
	Overall Level	1

Table 1: Security Levels

Public Material – May be reproduced only in its original entirety (without revision).

2 Cryptographic Module Specification

The ST33KTPM2X / ST33KTPM2XSPI / ST33KTPM2A / ST33KTPM2I module, hereafter denoted as the Module, is a fully integrated security module implementing revision 1.59 of the Trusted Computing Group (TCG) specification for Trusted Platform Modules (TPM) version 2.0.

2.1 Description

Purpose and Use:

The Module is intended for use by US Federal agencies or other markets that require FIPS 140-3 validated to security level 1. The Module is designed to be integrated into personal computers or any other embedded electronic systems. TPM is primarily used for cryptographic keys generation, keys storage, keys management and secure storage for digital certificates.

Module Type: Hardware

Module Embodiment: Single Chip

Module Characteristics:

Cryptographic Boundary:

The cryptographic boundary of the Module is defined as the perimeter of the IC package and is represented in the next figure. The Module is composed of:

- Two CPU cores, each including an MPU.
- Memories (RAMs, Flash and ROM) that store data or FW.
- HW accelerators for CRC (16 and 32-bits), symmetric cryptographic operations (AES) and asymmetric cryptographic operations (RSA/ECC).
- A clock generator and timers.
- An entropy source covered by an ESV Certificate.
- SPI and I^2C master/slave blocks.
- An administration block dedicated to chip security configuration and alarms detection.

Public Material – May be reproduced only in its original entirety (without revision).

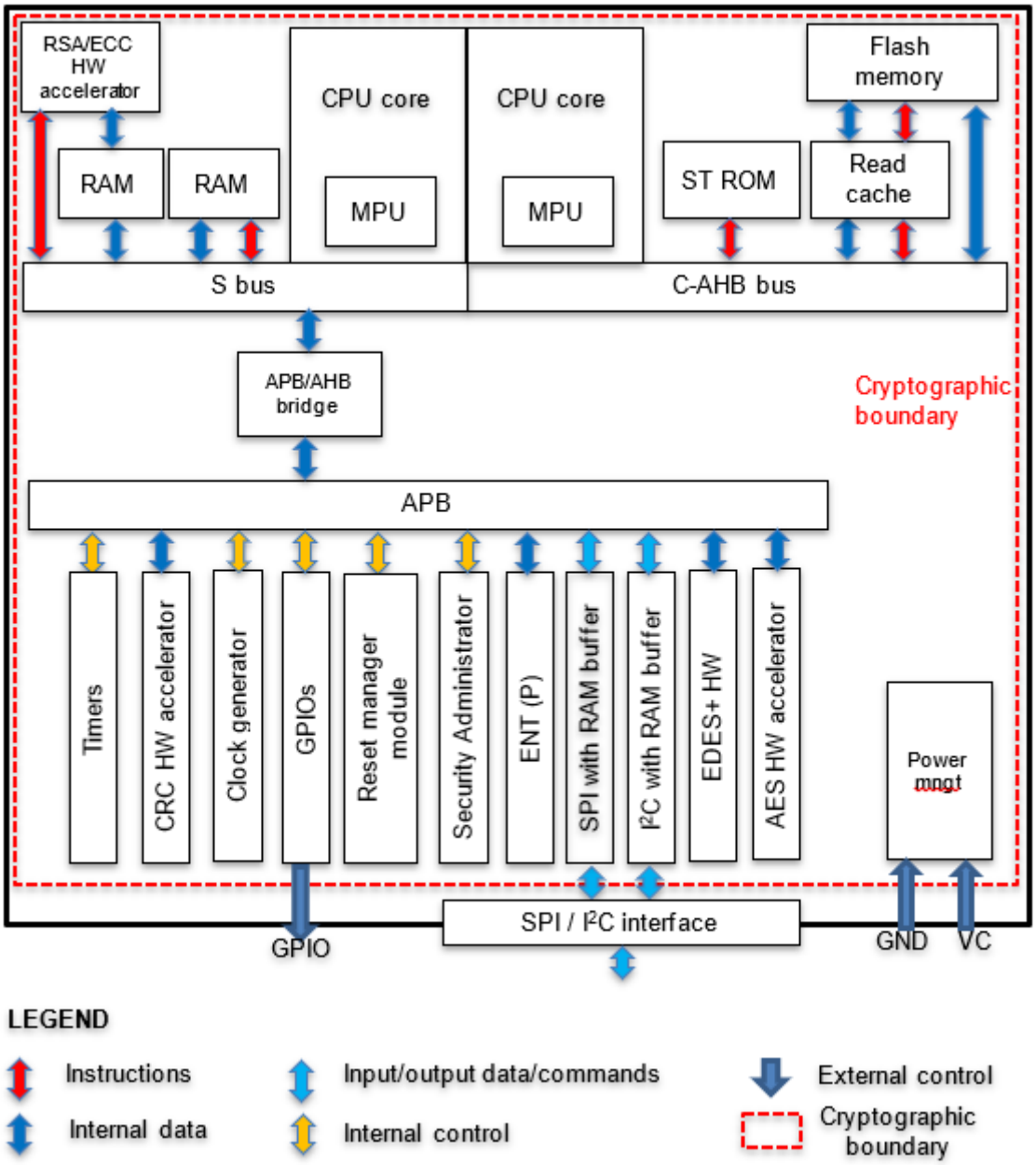


Figure 1 – HW block diagram

Public Material – May be reproduced only in its original entirety (without revision).

2.2 Tested and Vendor Affirmed Module Version and Identification

Tested Module Identification – Hardware:

The operating environments covered by the FIPS 140-3 validation are summarized in the table below:

Model and/or Part Number	Hardware Version	Firmware Version	Processors	Features
ST33KTPM2X	ST33K1M5T revC & revD	9.257 (dec.) 0x00.09.01.01 (hex.)	ST33K1M5T	SPI or I2C The interface is exclusive and selectable dynamically during product boot.
ST33KTPM2XSPI	ST33K1M5T revC & revD	9.257 (dec.) 0x00.09.01.01 (hex.) or 9.258 (dec.) 0x00.09.01.02 (hex.)	ST33K1M5T	SPI
ST33KTPM2A	ST33K1M5A revB	10.257 (dec.) 0x00.0A.01.01 (hex.)	ST33K1M5A	SPI or I2C The interface is exclusive and selectable dynamically during product boot.
ST33KTPM2I	ST33K1M5A revB	10.257 (dec.) 0x00.0A.01.01 (hex.)	ST33K1M5A	SPI or I2C The interface is exclusive and selectable dynamically during product boot.

Table 2: Tested Module Identification – Hardware

ST33KTPM2X and ST33KTPM2XSPI are manufactured in the UFQFPN32 package:

- UFQFPN32
  - Ultra-thin pitch Quad Flat No-lead 32-pin
  - 5 x 5 mm

Public Material – May be reproduced only in its original entirety (without revision).

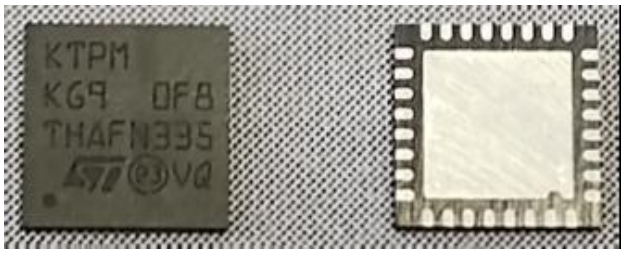


Figure 2 – UFQFPN32 Package

ST33KTPM2A and ST33KTPM2I are manufactured in the UFQFPN32 WF package:

- UFQFPN32 WF
  - Ultra-thin pitch Quad Flat No-lead 32-pin Wettable Flanks
  - 5 x 5 mm

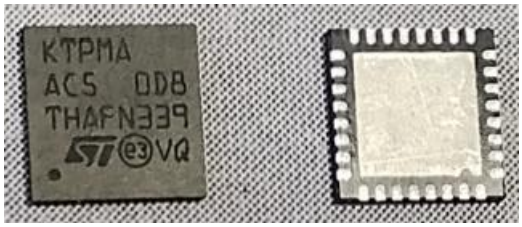


Figure 3 – UFQFPN32 WF Package

The ST33KTPM2A product is also manufactured in the TSSOP20 package:

- TSSOP 20-pin
- 6.5 x 4.4 mm

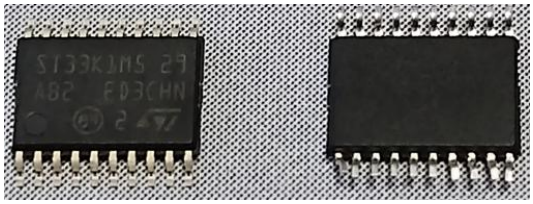


Figure 4 – TSSOP20 Package

The ST33KTPM2I product is also manufactured in the WLCSP24 package:

- WLCSP 24-pin
- 1.8 x 2.5 mm



Figure 5 – WLCSP24 Package

Public Material – May be reproduced only in its original entirety (without revision).

Products list:

The Module configurations indicated in Section 2.2 are defined into several manufactured products listed hereafter.

The default firmware version of KE2 is 9.256. To operate with firmware version 9.257, module must be first field upgraded to 9.257.

	Module Configuration*
Module name / HW P/N	ST33KTPM2XSPI
Package	UFQFPN32
Interface	SPI
Marking	KTPM KE2
FW version	00.09.01.01 (9.257)
TPM2.0 revision	1.59

Table 3 – KE2 Module Configuration with FW 9.257

To operate with firmware version 9.258, module must be first field upgraded to 9.258.

	Module Configuration*
Module name / HW P/N	ST33KTPM2XSPI
Package	UFQFPN32
Interface	SPI
Marking	KTPM KE2
FW version	00.09.01.02 (9.258)
TPM2.0 revision	1.59

Table 4 – KE2 Module Configuration with FW 9.258

The default firmware version of KE3 is 9.256. To operate with firmware version 9.257, module must be first field upgraded to 9.257.

	Module Configuration*
Module name / HW P/N	ST33KTPM2X
Package	UFQFPN32
Interface	SPI / I^2C
Marking	KTPM KE3
FW version	00.09.01.01 (9.257)
TPM2.0 revision	1.59

Table 5 – KE3 Module Configuration with FW 9.257

Public Material – May be reproduced only in its original entirety (without revision).

	Module Configuration*
Module name / HW P/N	ST33KTPM2XSPI
Package	UFQFPN32
Interface	SPI
Marking	KTPM KG8
FW version	00.09.01.01 (9.257)
TPM2.0 revision	1.59

Table 6 – KG8 Module Configuration

The default firmware version of KG8 is 9.257. To operate with firmware version 9.258, module must be first field upgraded to 9.258.

	Module Configuration*
Module name / HW P/N	ST33KTPM2XSPI
Package	UFQFPN32
Interface	SPI
Marking	KTPM KG8
FW version	00.09.01.02 (9.258)
TPM2.0 revision	1.59

Table 7 – KG8 Module Configuration with FW 9.258

	Module Configuration*
Module name / HW P/N	ST33KTPM2X
Package	UFQFPN32
Interface	SPI / I^2C
Marking	KTPM KG9
FW version	00.09.01.01 (9.257)
TPM2.0 revision	1.59

Table 8 – KG9 Module Configuration

	Module Configuration*
Module name / HW P/N	ST33KTPM2XSPI
Package	UFQFPN32
Interface	SPI
Marking	KTPM KJ5

Public Material – May be reproduced only in its original entirety (without revision).

FW version	00.09.01.02 (9.258)
TPM2.0 revision	1.59

Table 9 – KJ5 Module Configuration

	Module Configuration*
Module name / HW P/N	ST33KTPM2I
Package	UFQFPN32 WF WLCSP24
Interface	SPI
Marking	KTPM ZA9
FW version	00.0A.01.01 (10.257)
TPM2.0 revision	1.59

Table 10 – ZA9 Module Configuration

	Module Configuration*
Module name / HW P/N	ST33KTPM2A
Package	UFQFPN32 WF TSSOP20
Interface	SPI
Marking	KTPM AC5
FW version	00.0A.01.01 (10.257)
TPM2.0 revision	1.59

Table 11 – AC5 Module Configuration

The current FIPS 140-3 level 1 Security Policy applies to the Module configurations listed above. For the configurations supporting both SPI and I^2C interfaces, the selection of the mode is done during the boot of the Module.

2.3 Excluded Components

N/A

2.4 Modes of Operation

Modes List and Description:

Mode Name	Description	Types of Services	Status Indicator
Normal mode	TPM is in normal operation mode	Approved	TPM2_GetCapability (capability = TPM_CAP_VENDOR_PROPERTIES) with the

Public Material – May be reproduced only in its original entirety (without revision).

Mode Name	Description	Types of Services	Status Indicator
	when all pre-operational and conditional self-tests (apart from FW load and PCT tests) are complete. All approved services are usable. The corresponding indicator reports if the service uses an approved cryptographic algorithm or security function.		sub-capability TPM_SUBCAP_VENDOR_TPMA_MODES = 0x7 shall be used. It outputs a 2-bit indicator equals to 01b if the module is in an approved mode of operation
Non-approved mode of operation	The module enters a non-approved mode if one of the non-approved services is used by the operator.	Non-Approved	TPM2_GetCapability (capability = TPM_CAP_VENDOR_PROPERTIES) with the sub-capability TPM_SUBCAP_VENDOR_TPMA_MODES = 0x7 shall be used. It outputs a 2-bit indicator equals to 10b if the module is in a non-approved mode of operation

Table 12: Modes List and Description

2.5 Algorithms

Approved Algorithms:

The Module implements the Approved cryptographic algorithms listed in the table below.

Algorithm	CAVP Cert	Properties	Reference
AES-CBC	A2553	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CFB128	A2553	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-CTR	A2553	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
AES-ECB	A2553	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A

Public Material – May be reproduced only in its original entirety (without revision).

Algorithm	CAVP Cert	Properties	Reference
AES-OFB	A2553	Direction - Decrypt, Encrypt Key Length - 128, 192, 256	SP 800-38A
ECDSA KeyGen (FIPS186-4)	A2555	Curve - P-256, P-384 Secret Generation Mode - Extra Bits	FIPS 186-4
ECDSA KeyVer (FIPS186-4)	A2555	Curve - P-256, P-384	FIPS 186-4
ECDSA SigGen (FIPS186-4)	A2555	Component - No Curve - P-256, P-384 Hash Algorithm - SHA2-256, SHA2-384, SHA3-256, SHA3-384	FIPS 186-4
ECDSA SigVer (FIPS186-4)	A2555	Component - No Curve - P-256, P-384 Hash Algorithm - SHA2-256, SHA2-384, SHA3-256, SHA3-384	FIPS 186-4
Hash DRBG	A2547	Prediction Resistance - No Mode - SHA2-256	SP 800-90A Rev. 1
HMAC-SHA-1	A2551	Key Length - Key Length: 8-8192 Increment 8	FIPS 198-1
HMAC-SHA-1	A2552	Key Length - Key Length: 8-8192 Increment 8	FIPS 198-1
HMAC-SHA2-256	A2551	Key Length - Key Length: 8-8192 Increment 8	FIPS 198-1
HMAC-SHA2-256	A2552	Key Length - Key Length: 8-8192 Increment 8	FIPS 198-1
HMAC-SHA2-384	A2551	Key Length - Key Length: 8-8192 Increment 8	FIPS 198-1
HMAC-SHA2-384	A2552	Key Length - Key Length: 8-8192 Increment 8	FIPS 198-1
HMAC-SHA3-256	A2551	Key Length - Key Length: 8-8192 Increment 8	FIPS 198-1
HMAC-SHA3-256	A2552	Key Length - Key Length: 8-8192 Increment 8	FIPS 198-1
HMAC-SHA3-384	A2551	Key Length - Key Length: 8-8192 Increment 8	FIPS 198-1
HMAC-SHA3-384	A2552	Key Length - Key Length: 8-8192 Increment 8	FIPS 198-1
KAS-ECC Sp800-56Ar3	A2555	Domain Parameter Generation Methods - P-256, P-384 Function - Full Validation, Key Pair Generation Scheme - fullUnified - KAS Role - Initiator, Responder KDF Methods -	SP 800-56A Rev. 3

Public Material – May be reproduced only in its original entirety (without revision).

Algorithm	CAVP Cert	Properties	Reference
		oneStepKdf - Key Length - 128 onePassDh - KAS Role - Initiator, Responder KDF Methods - oneStepKdf - Key Length - 128	
KDF SP800-108	A2550	KDF Mode - Counter Supported Lengths - Supported Lengths: 160-512 Increment 8	SP 800-108 Rev. 1
KTS-IFC	A2554	Modulo - 2048, 3072, 4096 Key Generation Methods - rsakpg1-basic Scheme - KTS-OAEP-basic - KAS Role - initiator, responder Key Transport Method - Key Length - 256	SP 800-56B Rev. 2
RSA KeyGen (FIPS186-4)	A2554	Key Generation Mode - B.3.3 Modulo - 2048, 3072, 4096 Primality Tests - Table C.3 Private Key Format - Standard	FIPS 186-4
RSA SigGen (FIPS186-4)	A2554	Signature Type - PKCS 1.5, PKCSPSS Modulo - 2048, 3072, 4096	FIPS 186-4
RSA SigVer (FIPS186-4)	A2554	Signature Type - PKCS 1.5, PKCSPSS Modulo - 1024, 2048, 3072, 4096	FIPS 186-4
SHA-1	A2548	Message Length - Message Length: 160, 0-65528 Increment 8	FIPS 180-4
SHA-1	A2549	Message Length - Message Length: 160, 0-65528 Increment 8	FIPS 180-4
SHA2-256	A2548	Message Length - Message Length: 256, 0-65528 Increment 8	FIPS 180-4
SHA2-256	A2549	Message Length - Message Length: 256, 0-65528 Increment 8	FIPS 180-4
SHA2-384	A2548	Message Length - Message Length: 384, 0-65528 Increment 8	FIPS 180-4
SHA2-384	A2549	Message Length - Message Length: 384, 0-65528 Increment 8	FIPS 180-4
SHA3-256	A2548	Message Length - Message Length: 0-65528 Increment 8	FIPS 202
SHA3-384	A2548	Message Length - Message Length: 0-65528 Increment 8	FIPS 202

Table 13: Approved Algorithms

Public Material – May be reproduced only in its original entirety (without revision).

Vendor-Affirmed Algorithms:

The Module implements the Vendor Affirmed cryptographic algorithms listed.

Name	Properties	Implementation	Reference
CKG	Key Type:Symmetric and Asymmetric	N/A	NIST SP 800-133r2, Section 4 example 1, Sections 5.1, 5.2, and 6.1

Table 14: Vendor-Affirmed Algorithms

Non-Approved, Allowed Algorithms:

N/A for this module.

Non-Approved, Allowed Algorithms with No Security Claimed:

The Module implements the Non-Approved, Allowed cryptographic Algorithms with No Security Claimed.

Name	Caveat	Use and Function
XOR	No security claimed per IG 2.4.A with the example of scenario #1. The algorithm is not used except for this purpose, does not access or share CSPs in a way that counters the requirements of the IG, not intended to be used as a security function, and can't be confused for a security function.	Obfuscation of input or output data

Table 15: Non-Approved, Allowed Algorithms with No Security Claimed

Non-Approved, Not Allowed Algorithms:

The Module implements the Non-Approved, Not Allowed cryptographic algorithms listed.

Name	Use and Function
ECC BN P-256 (non-compliant)	Key generation, digital signature generation based on ECC BN P-256
ECC derived keys (non-compliant)	Secret exchange or digital signature generation/verification
ECDAA (non-compliant)	Key generation, digital signature generation
ECDSA (non-compliant)	Digital signature with an ECC signing key generated with an undetermined scheme (field inPublic.buffer.parameters.scheme.scheme = TPM_ALG_NULL), derived from a derivation parent key, or a key loaded in the NULL hierarchy

Public Material – May be reproduced only in its original entirety (without revision).

Name	Use and Function
ECSchnorr (non-compliant)	Key generation, digital signature generation and verification
HMAC (non-compliant)	Key length < 112 bits for message authentication
KAS (non-compliant)	Key agreement with an ECC key that has an undetermined scheme (field inPublic.buffer.parameters.scheme.scheme = TPM_ALG_NULL)
KBKDF (non-compliant)	Non-Approved key derivation usage
KTS-IFC (non-compliant)	Key encapsulation with an RSA decryption key that has an undetermined scheme (field inPublic.buffer.parameters.scheme.scheme = TPM_ALG_NULL)
RSA (non-compliant)	1024-bit RSA digital signature generation or with a key loaded in the Null hierarchy
RSA with no padding mode (null scheme) (non-compliant)	Key transport
RSAES-PKCS1-v1_5 (non-compliant)	Key transport
SHA-1 (non-compliant)	Digital signature generation

Table 16: Non-Approved, Not Allowed Algorithms

2.6 Security Function Implementations

Next table shows the Security Function Implementations that the Module implements:

Name	Type	Description	Properties	Algorithms
AES-DEC	BC-UnAuth	Unauthenticated Decryption	Publication:NIST FIPS 197	AES-CBC: (A2553) Direction: Encrypt, Decrypt Key Length: 128, 192, 256 AES-CFB128: (A2553) Direction: Encrypt, Decrypt Key Length: 128, 192, 256 AES-CTR: (A2553) Direction:

Public Material – May be reproduced only in its original entirety (without revision).

Name	Type	Description	Properties	Algorithms
				Encrypt, Decrypt Key Length: 128, 192, 256 AES-ECB: (A2553) Direction: Encrypt, Decrypt Key Length: 128, 192, 256 AES-OFB: (A2553) Direction: Encrypt, Decrypt Key Length: 128, 192, 256
AES-ENC	BC-UnAuth	Unauthenticated Encryption	Publication:NIST FIPS 197	AES-CBC: (A2553) Directions: Encrypt, Decrypt Key Lengths: 128, 192, 256 AES-CFB128: (A2553) Directions: Encrypt, Decrypt Key Lengths: 128, 192, 256 AES-CTR: (A2553) Directions: Encrypt, Decrypt Key Lengths: 128, 192, 256 AES-ECB: (A2553) Directions: Encrypt, Decrypt Key Lengths: 128, 192, 256 AES-OFB: (A2553) Direction: Encrypt, Decrypt Key Lengths: 128, 192, 256

Public Material – May be reproduced only in its original entirety (without revision).

Name	Type	Description	Properties	Algorithms
DRBG	DRBG	Random Number Generation	Publications:NIST SP 800-90A, FIPS 180-4	Hash DRBG: (A2547) Prediction Resistance: No Mode: SHA2-256 SHA2-256: (A2548) Message Length: 256, 0-65528 Increment 8
ENT-ESV	ENT-ESV	ESV	Publication:NIST SP 800-90B	SHA2-256: (A2548) Message Length: 256, 0-65528 Increment 8
KAS	KAS-Full	Key establishment	Publications:NIST SP 800-56Ar3 IG:IG D.F. Scenario 2, path (2), end-to-end Key confirmation:no Key derivation:KDA (tested as part of KAS certificate) Caveat:Key establishment methodology provides 128 bits of security strength	KAS-ECC Sp800-56Ar3: (A2555) Domain Parameter Generation Methods: P-256, P-384 Functions: Key Pair Generation, Full Validation KAS Roles: Initiator, Responder Key Length: 128 SHA-1: (A2548, A2549) Message Length: 160, 0-65528 Increment 8 SHA2-256: (A2548, A2549) Message Length: 256, 0-65528 Increment 8 SHA2-384: (A2548, A2549) Message Length: 384, 0-65528

Public Material – May be reproduced only in its original entirety (without revision).

Name	Type	Description	Properties	Algorithms
				Increment 8 SHA3-256: (A2548) Message Length: 0-65528 Increment 8 SHA3-384: (A2548) Message Length: 0-65528 Increment 8
KAS-KeyGen	KAS-KeyGen	KAS-ECC Key Generation	Publication:NIST SP 800-56Arev3	KAS-ECC Sp800-56Ar3: (A2555) Domain Parameter Generation Methods: P-256, P-384 Functions: Key Pair Generation, Full Validation KAS Roles: Initiator, Responder Key Length: 128 CKG: () Key Type: Asymmetric
KBKDF	KBKDF	Key-Based Key Derivation	Publication:NIST SP 800-108	KDF SP800-108: (A2550) KDF Mode: Counter Supported Lengths: 160-512 Increment 8 SHA-1: (A2548) Message Length: 160, 0-65528 Increment 8 SHA2-256: (A2548) Message Length: 256, 0-65528 Increment 8

Public Material – May be reproduced only in its original entirety (without revision).

Name	Type	Description	Properties	Algorithms
				SHA2-384: (A2548) Message Length: 384, 0-65528 Increment 8 SHA3-256: (A2548) Message Length: 0-65528 Increment 8 SHA3-384: (A2548) Message Length: 0-65528 Increment 8
KTS-Decap	KTS-Decap	Key Decapsulation	Publication:NIST SP 800-56Br2, IG D.G IG D.G:Approved method Caveat:Key establishment methodology provides between 112 and 150 bits of security strength	KTS-IFC: (A2554) Modulo: 2048, 3072, 4096 Key Generation Methods: rsakpg1-basic KAS Roles: initiator, responder Key Length: 256
KTS-Encap	KTS-Encap	Key Encapsulation	Publication:NIST SP 800-56Br2, IG D.G IG D.G:Approved method Caveat:Key establishment methodology provides between 112 and 150 bits of security strength	KTS-IFC: (A2554) Modulo: 2048, 3072, 4096 Key Generation Methods: rsakpg1-basic KAS Roles: initiator, responder Key Length: 256
KTS-Unwrap	KTS-Unwrap	Key transport	Publication:NIST SP 800-38F IG D.G:Approved method Caveat:Key establishment methodology provides	HMAC-SHA2-256: (A2551) Key Length: 8-8192 Increment 8 AES-CFB128: (A2553) Direction:

Public Material – May be reproduced only in its original entirety (without revision).

Name	Type	Description	Properties	Algorithms
			128, 192 or 256 bits of security strength	Encrypt, Decrypt Key Length: 128, 192, 256
KTS-Wrap	KTS-Wrap	Key transport	Publication:NIST SP 800-38F IG D.G:Approved method Caveat:Key establishment methodology provides 128, 192 or 256 bits of security strength	HMAC-SHA2-256: (A2551) Key Length: 8-8192 Increment 8 AES-CFB128: (A2553) Direction: Encrypt, Decrypt Key Length: 128, 192, 256
KeyGen	AsymKeyPair-KeyGen	Key-Pair Generation	Publication:NIST FIPS 186-4	ECDSA KeyGen (FIPS186-4): (A2555) Curves: P-256, P-384 Secret Generation Mode: Extra Bits RSA KeyGen (FIPS186-4): (A2554) Key Generation Mode: B.3.3 Modulo: 2048, 3072, 4096 Primality Tests: Table C.3 Private Key Format: Standard CKG: () Key Type: Asymmetric
KeyVal	AsymKeyPair-PubKeyVal	Public Key Validation	Publications:NIST SP 800-56Ar3, SP 800-56Br2, FIPS 186-5	KAS-ECC Sp800-56Ar3: (A2555) Domain Parameter Generation Methods: P-256, P-384

Public Material – May be reproduced only in its original entirety (without revision).

Name	Type	Description	Properties	Algorithms
				Functions: Key Pair Generation, Full Validation KAS Roles: Initiator, Responder Key Length: 128 KTS-IFC: (A2554) Modulo: 2048, 3072, 4096 Key Generation Methods: rsakpg1-basic KAS Roles: initiator, responder Key Length: 256
KeyVer	AsymKeyPair-KeyVer	Key-Pair Verification	Publication:NIST FIPS 186-4	ECDSA KeyVer (FIPS186-4): (A2555) Curves: P-256, P-384
MAC	MAC	Message Authentication	Publications:NIST FIPS 198, FIPS 180-4, FIPS 202	HMAC-SHA-1: (A2551, A2552) Key Length: 8-8192 Increment 8 HMAC-SHA2-256: (A2551, A2552) Key Length: 8-8192 Increment 8 HMAC-SHA2-384: (A2551, A2552) Key Length: 8-8192 Increment 8 HMAC-SHA3-256: (A2551, A2552) Key Length: 8-8192 Increment

Public Material – May be reproduced only in its original entirety (without revision).

Name	Type	Description	Properties	Algorithms
				8 HMAC-SHA3-384: (A2551, A2552) Key Length: 8-8192 Increment 8 SHA-1: (A2548) Message Length: 160, 0-65528 Increment 8 SHA2-256: (A2548) Message Length: 256, 0-65528 Increment 8 SHA2-384: (A2548) Message Length: 384, 0-65528 Increment 8 SHA3-256: (A2548) Message Length: 0-65528 Increment 8 SHA3-384: (A2548) Message Length: 0-65528 Increment 8
SHA	SHA	Secure Hash	Publications:NIST FIPS 180-4, FIPS 202	SHA-1: (A2548, A2549) Message Length: 160, 0-65528 Increment 8 SHA2-256: (A2548, A2549) Message Length: 256, 0-65528 Increment 8 SHA2-384: (A2548, A2549) Message Length: 384, 0-65528

Public Material – May be reproduced only in its original entirety (without revision).

Name	Type	Description	Properties	Algorithms
				Increment 8 SHA3-256: (A2548) Message Length: 0-65528 Increment 8 SHA3-384: (A2548) Message Length: 0-65528 Increment 8
SKG	CKG	Symmetric Key Generation	Publications:NIST SP 800-133r2, IG D.H	Hash DRBG: (A2547) Prediction Resistance: No Mode: SHA2-256
SigGen	DigSig-SigGen	Signature Generation	Publications:NIST FIPS 186-4, FIPS 180-4, FIPS 202	ECDSA SigGen (FIPS186-4): (A2555) Component: No Curves: P-256, P-384 Hash Algorithm: SHA2-256, SHA2-384, SHA3-256, SHA3-384 RSA SigGen (FIPS186-4): (A2554) Signature Types: PKCS 1.5, PKCSPSS Modulo: 2048, 3072, 4096 SHA2-256: (A2548) Message Length: 256, 0-65528 Increment 8 SHA2-384: (A2548) Message Length: 384, 0-65528

Public Material – May be reproduced only in its original entirety (without revision).

Name	Type	Description	Properties	Algorithms
				Increment 8 SHA3-256: (A2548) Message Length: 0-65528 Increment 8 SHA3-384: (A2548) Message Length: 0-65528 Increment 8
SigVer	DigSig-SigVer	Signature Verification	Publications:NIST FIPS 186-4, FIPS 180-4, FIPS 202	ECDSA SigVer (FIPS186-4): (A2555) Component: No Curves: P-256, P-384 Hash Algorithm: SHA2-256, SHA2-384, SHA3-256, SHA3-384 RSA SigVer (FIPS186-4): (A2554) Signature Types: PKCS 1.5, PKCSPSS Modulo: 1024, 2048, 3072, 4096 SHA2-256: (A2548) Message Length: 256, 0-65528 Increment 8 SHA2-384: (A2549) Message Length: 384, 0-65528 Increment 8 SHA3-256: (A2548) Message Length: 0-65528

Public Material – May be reproduced only in its original entirety (without revision).

Name	Type	Description	Properties	Algorithms
				Increment 8 SHA3-384: (A2548) Message Length: 0-65528 Increment 8

Table 17: Security Function Implementations

2.7 Algorithm Specific Information

FIPS 186-4 Transition

For ECDSA KeyGen (FIPS 186-4) (Cert. #A2555), ECDSA KeyVer (FIPS 186-4) (Cert. #A2555), ECDSA SigGen (FIPS 186-4) (Cert. #A2555), ECDSA SigVer (FIPS 186-4) (Cert. #A2555), RSA KeyGen (FIPS 186-4) (Cert. #A2554), RSA SigGen (FIPS 186-4) (Cert. #A2554), and RSA SigVer (FIPS 186-4) (Cert. #A2554), IG C.K Additional Comment 3 is applicable. Specifically, the validation date of Apr 25, 2022 preceded Feb 5, 2024. Additionally, the CAVP tests conducted for the said certificates are mathematically identical to the FIPS 186-5 CAVP tests.

Compliance to IG C.A

The module's use of elliptic curves conforms to IG C.A. The Security Functions Implementation Table lists the approved curves.

Compliance to IG C.M

Algorithms designated as "Legacy" can only be used on data that was generated prior to the Legacy Date specified in FIPS 140-3 IG C.M. For RSA signature verification using 1024-bit keys, the Legacy Date specified by IG C.M is 2011.

Compliance to IG D.M

All of the key derivation keys generated through KDF SP 800-108 (Cert. #A2550) are derived using an approved method, namely, Hash DRBG (Cert. #A2550) or CKG per NIST SP 800-133r2, in the approved mode. The encoding of the data fields (i.e., Label, Context, and Length L) is complaint to Section 6.4 of NIST SP 800-133r1. No asymmetric keys are generated through KBKDF.

Compliance to NIST SP 800-56Ar3 Assurances

For KAS ECC, the module satisfies IG D.F Scenario 2 path (2). The key derivation functions comply with NIST SP 800-56Cr2 (i.e., OneStep KDF). Furthermore, the module obtains the appropriate assurances as required in Section 5.6.2 of NIST SP 800-56Ar3. Full public key validations are performed at every key generation and whenever a static key pair is received (NIST SP 800-56Ar3 Section 5.6.2.3.3). No key confirmation is implemented.

Compliance to NIST SP 800-56Br2 Assurances

For KTS RSA, the module satisfies IG D.G. The RSA keys for KTS-IFC (Cert. #A2554) have lengths of at least 2048 bits, which are acceptable per NIST SP 800-131Ar2. The modulus sizes, the method for key generation, and the roles (initiator, i.e., the module performs encapsulation, or responder, i.e., the

Public Material – May be reproduced only in its original entirety (without revision).

module performs unencapsulation) are specified in the Security Functions Implementation Table. The RSA key generation algorithm is CAVP validated (Cert. #A2554). No key confirmation is implemented.

The module implements the KTS-OAEP key transport scheme and obtains the appropriate assurances as required in Sections 5 and 6 of NIST SP 800-56Br2. The module performs partial public-key validation. No key confirmation is implemented

Limited use of SHA-1

The use of SHA-1 (Certs. #A2548, #A2549) is restricted within the module to non-digital signature applications.

2.8 RBG and Entropy

The Module implements:

- A Hash-DRBG (Cert. #A2547) based on SHA2-256 and compliant with the [90A] standard (state is referred as drbgState in SSPs table). It is seeded at each module start-up with 512 bits generated from the internal, physical entropy source (ESV Cert. #E41). The entropy source provides 0.8164 bits of entropy per bit sampled, which results in a total of 418 bits of entropy (i.e., 512 * 0.81640625) being used to instantiate the DRBG. This is sufficient to claim a full security strength of 256 bits.
- A transient Hash-DRBG based on SHA2-256 and compliant with the [90A] standard (state is referred as tdrbgState in SSPs table) involved only in primary keys generation and seeded as defined in [TPM2.0 Part1] and [TPM2.0 Part3].
- An entropy source as detailed below:

Cert Number	Vendor Name
E41	STMicroelectronics

Table 18: Entropy Certificates

Name	Type	Operational Environment	Sample Size	Entropy per Sample	Conditioning Component
Trusted Platform Module ST33KTPM2X, ST33KTPM2XSPI, ST33KTPM2XI2C, ST33KTPM2A, ST33KTPM2I entropy source	Physical	ST33K1M5T/A platforms	256 bit	209 bits	A2548 (SHA2-256)

Table 19: Entropy Sources

2.9 Key Generation

Public Material – May be reproduced only in its original entirety (without revision).

For Key Generation methods, see Section 2.6 Security Function Implementations above.

2.10 Key Establishment

Key Agreement Information

For Key Establishment methods, see Section 2.6 Security Function Implementations above.

Key Transport Information

For Key Transport methods, see Section 2.6 Security Function Implementations above.

2.11 Industry Protocols

The Module does not implement any Industry Protocols.

Public Material – May be reproduced only in its original entirety (without revision).

3 Cryptographic Module Interfaces

3.1 Ports and Interfaces

The Module's ports and associated FIPS-defined logical interface categories are listed below.

Physical Port	Logical Interface(s)	Data That Passes
SPI_NSS / SPI_CLK / SPI_MOSI / I2C_SCL / I2C_SDA	Data Input	Data (plaintext data, ciphertext data and SSPs) provided to the security module as part of an input processing command
SPI_NSS / SPI_CLK / SPI_MISO / I2C_SCL / I2C_SDA	Data Output	Data (plaintext data, ciphertext data and SSPs) output by the security module as part of the response to a processing command
SPI_NSS / SPI_CLK / SPI_MISO / I2C_SCL / I2C_SDA / PIRQ	Status Output	Status output by the security module (responseCode parameter of a response)
SPI_NSS / SPI_CLK / SPI_MISO / I2C_SCL / I2C_SDA / PIRQ	Control Output	Control parts of the TPM responses output by the security module. It concerns all bytes of a response except plaintext data, ciphertext data and SSPs (output with the data output interface) and except the responseCode of a response (output with the status output interface)
SPI_NSS / SPI_CLK / SPI_MOSI / I2C_SCL / I2C_SDA / RESET / PP	Control Input	Control parts of the TPM commands provided to the security module. It concerns all bytes of a command except plaintext data, ciphertext data and SSPs (entered with the data input interface).
VCC / GND	Power	Power interface of the security module

Table 20: Ports and Interfaces

Additional details concerning the ports and interfaces of TPM:

1. Control and data inputs are multiplexed over the same physical interface. Control and data are distinguished by properly parsing input TPM command parameters according to input structures description, indicated for each command in [TPM2.0 Part3]. Some commands only deal with control input and status output parameters.

Public Material – May be reproduced only in its original entirety (without revision).

2. Status, data and control output are multiplexed over the same physical interface. Status, data and control are distinguished by properly setting output TPM response parameters according to output structures description, indicated for each command in [TPM2.0 Part3].
3. The logical state machine and the command structure parsing of the module prevent from using input data externally from the “data input path” and prevent from outputting data externally from the “data output path”.
4. While performing key generation or key zeroisation (no manual key entry on TPM), the output data path is logically disconnected while the output status path remains connected to report any possible failure during command processing. Generally, the output data path is only connected when TPM outputs response containing data.
5. To prevent the inadvertent output of CSPs in plaintext form on TPM2_Duplicate, the two following independent internal actions are performed:
    a. Verification of the encryptedDuplication attribute of the key to be duplicated
    b. Verification of the handle of the new parent of the key to be duplicated

    encryptedDuplication attribute must be set to 0 and new handle must be set to the null handle to authorize outputting the private part of the key in plaintext form.
6. The logical state machine and command structure of the module guarantees the inhibition of all data output via the data output interface whenever an error state exists and while doing self-tests.
7. The status output interface remains active during the error state to output the status of the security module with the service TPM2_GetCapability and TPM2_GetTestResult.

3.2 Pinout description

The ST33KTPM2X / ST33KTPM2A / ST33KTPM2I security modules support both SPI and I^2C physical interfaces but only one interface is configured during TPM boot. The interface configured remains active until the next module reset.

The ST33KTPM2XSPI security module only supports the SPI physical interface.

The table below gives a description of the product pins.

Signal	Type	Description*
VCC	Input	**Power supply**. This pin must be connected to 1.8V or 3.3V DC power rail supplied by the motherboard.
GND	Input	GND has to be connected to the main motherboard ground.
RESET	Input	Reset used to re-initialize the device
I2C SCL / GPIO5	Input or Input/Output	I^2C serial clock (Open drain with no weak pull-up resistor) or GPIO if SPI interface is selected
I2C SDA / GPIO6	Input/Output	I^2C serial data (Open drain with no weak pull-up resistor) or GPIO if SPI interface is selected
PIRQ	Output	IRQ used by TPM to generate an interrupt
SPI CLK / GPIO1	Input or Input/Output	SPI serial clock (output from master) or GPIO if I^2C interface is selected

Public Material – May be reproduced only in its original entirety (without revision).

SPI NSS / GPIO2	Input or Input/Output	SPI slave select (active low; output from master) or GPIO if I²C interface is selected
SPI MISO / GPIO0	Output or Input/Output	SPI Master Input, Slave Output (output from slave) or GPIO if I²C interface is selected
SPI MOSI / GPIO3	Input or Input/Output	SPI Master Output, Slave Input (output from master) or GPIO if I²C interface is selected
I2C SEL	Input	This pin must be connected to an external pull-down resistor to activate the I²C protocol during product boot time. It can remain unconnected for the SPI protocol. This pin is internal pull-up by default and becomes internal floating after I²C activation.
PP	Input	**Physical presence**, active high, internal pull-down. Used to indicate Physical Presence to the TPM.
NC	-	**Not Connected:** connected to the die but not usable. May be left unconnected. Internal pull-down.
NiC	-	**Not Internally Connected:** not connected to the die. May be left unconnected but no impact on TPM if connected.

Table 21 – Pins Definition

UFQFPN32 / UFQFPN32 WF configuration

The pin layout for the UFQFPN32 / UFQFPN32 WF package is shown in the next figure.

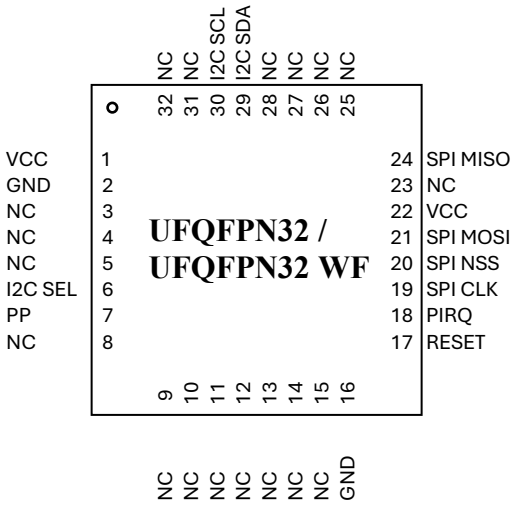


Figure 6 – UFQFPN32 / UFQFPN32 WF Pinout Diagram

TSSOP20 configuration

The pin layout for the TSSOP20 package is shown in the next figure.

Public Material – May be reproduced only in its original entirety (without revision).

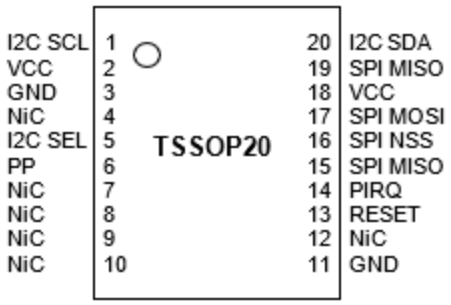


Figure 7 – TSSOP20 Pinout Diagram

WLCSP24 configuration

The pin layout for the WLCSP24 package is shown in the next figure.

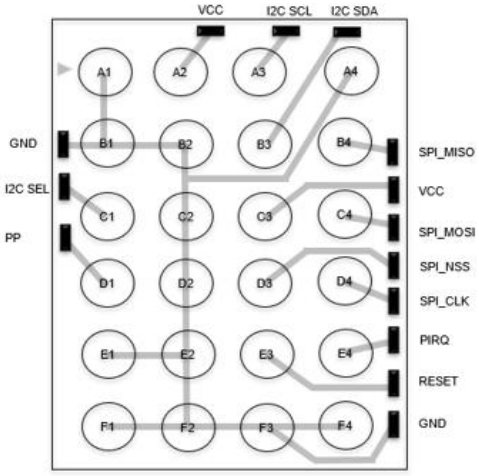


Figure 8 - WLCSP24 Pinout Diagram

Public Material – May be reproduced only in its original entirety (without revision).

4 Roles, Services, and Authentication

4.1 Authentication Methods

In the context of this FIPS 140-3 Security Level 1 evaluation, there is no authentication mechanism claimed to control access of the security module. The authorization mechanisms (password, HMAC and policy) provided by the TPM2.0 standard are available and protected as sensitive parameters but are not employed to satisfy FIPS 140-3 requirements. Crypto officer role is implicitly assumed by the operator when using services corresponding to that role.

N/A for this module.

4.2 Roles

The Roles Table below lists all operator roles supported by the Module.

Name	Type	Operator Type	Authentication Methods
Crypto officer (CO)	Role	Administrator of the Module	None
User (U)	Role	User of the Module	None
Unauthenticated	Role	Unauthenticated	None

Table 22: Roles

The Module does not provide a maintenance role or maintenance interface and does not support concurrent operators. The role is implicitly selected by the TPM operator on service execution by proving the knowledge of the enhanced authorization commands sequence and/or the authorization value of an object.

4.3 Approved Services

All services are accessible under the roles defined above and no specific access rights are considered to operate with keys and SSPs. Full services inputs and outputs are defined in [TPM2.0 Part3]. Next table indicates how mandatory services of [ISO/IEC 19790] (§7.4.3.1) are mapped to security module's services:

Mandatory service requested from [ISO/IEC 19790]*	Corresponding services from the security module
Show module's versioning information	TPM2_GetCapability
Show status	TPM2_GetTestResult
Perform self-tests	TPM2_SelfTest
Perform approved security functions	See approved services listed in next table
Perform zeroization	See services listed in section 9.3 SSP Zeroization Methods.

Public Material – May be reproduced only in its original entirety (without revision).

Table 23 – Mapping between services

All approved services implemented by the Module are listed in the table below:

The SSPs modes of access shown in the table below are defined as:

- G = Generate: The Module generates or derives the SSP.
- R = Read: The SSP is read from the Module (e.g., the SSP is output).
- W = Write: The SSP is updated, imported, or written to the Module (SSP is input).
- E = Execute: The Module uses the SSP in performing a cryptographic operation.
- Z = Zeroize: The Module zeroizes the SSP

Some details about information in the table:

- In "Name" column, **(I)** indicates that service is usable with sessions integrity mechanism, **(E)** indicates that service is usable with encryption session (encryption of 1st parameter of response), **(D)** indicates that service is usable with decryption session (decryption of 1st parameter of command)
- In "Indicator" column, the value of indicator can be approved, non-approved or non-security relevant.
- In "Inputs" column, commands inputs are not exhaustive, some non-security parameters are voluntarily missed. Full inputs of all commands are defined in **[TPM2.0 Part3]**
- In "Outputs" column, Outputs of all responses are defined in **[TPM2.0 Part3]**
- In "Security Function Implementation" column, security functions are referenced by their identifiers indicated in the tables of the paragraph 2.5.
- In "Roles" column, NA indicates that the service does not require authentication. The list of roles is indicated in paragraph 4.2.

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
Encryption mechanism provided by sessions	This service is not callable from TPM interface but is only used internally by any command and response with an encryption or decryption session. It consists in	Approved	Command or response	Encrypted parameter	AES - ENC AES - DEC DRBG KB KDF SHA SKG	Unauthenticated - sesSymKey: G,E,Z - drbgState: W,E - sesSalt: E - platform

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	decrypting the first parameter of a received command or encrypting the first parameter of a transmitted response.					Auth: E - endorsementAuth: E - ownerAuth: E - lockoutAuth: E - objAuth: E - nvAuth: E
Integrity mechanism provided by sessions	This service is not callable from TPM interface but is only used internally by any command and response with an authorization area. It consists in computing the integrity of the received command or transmitted response.	Approved	Command or response	Integrity value	MAC SHA	Unauthenticated - sesHmacKey: E,Z
TPM2_ActivateCredential (I/E/D)	Enables the association of a credential with an object in a way that ensures that the TPM has	Approved	Handle of the object with credentials, Handle	Decrypted certificate information	AES - DECKASKTS - Deca	Crypto officer (CO) - creSymKey: G,E,Z

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	validated the parameters of the credentialed object		of a loaded private key, Encrypted credential, Encrypted seed		pKTS - Unwrap KBKDFMAC SHA	- creHmacKey: G,E,Z - objSens: E - creSeed: W,E,Z - eccParams256: E - eccParams384: E - intSharedSecret: G,E,Z - extPubEccKey: W,E,Z
TPM2_Certify (I/E/D)	Proves that an object with a specific Name is loaded in the TPM	Approved	Handle of the object to certify, Handle of a signing key, Qualifying data for Signature scheme	Certification structure, Signature over the certification structure	SigGen DRBG KBKDFMAC SHA	User (U) - drbgState: W,E - objSens: E - shProof: E
TPM2_CertifyCreation (I/E/D)	Proves the association between an	Approved	Handle of the object to	Certification structure, Signature	SigGen DRB	User (U) - drbgStat

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	object and its creation data		certify, Handle of a signing key, Qualifying data for Signature scheme, Ticket Creation hash	over the certification structure	GKB KDFMAC SHA	e: W,E - objSens: E - nullProof: E - phProof: E - ehProof: E - shProof: E
TPM2_CertifyX509 (I/E/D)	X.509 certificate generation	Approved	Handle of the object to certify, Handle of a signing key, Partial certificate for Signature scheme	Additional certificate information Digest, Signature over the digest	SigGen SHA	User (U) - drbgState: W,E - objSens: E
TPM2_ChangeEPS (I)	Replaces the current endorsement primary seed (EPS, also known as ehSeed) with a value from the RNG and sets endorsementPolicy to the default	Approved	Authorization handle	None	DRBG	Crypto officer (CO) - drbgState: W,E - ehSeed: Z - ehProof: Z -

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	initialization value					objSeed: Z - objSens: Z - objPub: Z - ekRsa: Z - ekEcc: Z
TPM2_ChangePPS (I)	Replaces the current platform primary seed (PPS, also known as phSeed) with a value from the RNG and sets platformPolicy to the default initialization value	Approved	Authorization handle	None	DRBG	Crypto officer (CO) - drbgState: W,E - phProof: Z - phSeed: Z - objSeed: Z - objSens: Z - objPub: Z
TPM2_Clear (I)	Removes all TPM context associated with a specific Owner	Approved	Authorization handle	None	DRBG	Crypto officer (CO) - drbgState: W,E - shSeed: Z - ehProof:

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Z - shProof: Z - shProofForReseed: Z - objSeed: Z - objSens: Z - objPub: Z - objAuth: Z
TPM2_ClearControl (I)	Disables and enables the execution of TPM2_Clear()	Non-security relevant	Authorization handle, Set or clear disableOwnerFlag	None	None	Crypto officer (CO)
TPM2_ClockRateAdjust (I)	Adjusts the rate of advance of Clock and Time	Non-security relevant	Authorization handle, Clock update rate adjustment	None	None	Crypto officer (CO)
TPM2_ClockSet (I)	Advances the value of the TPM's clock	Non-security relevant	New time	None	None	Crypto officer (CO)

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_ContextLoad	Reloads a context that has been saved by TPM2_ContextSave()	Approved	Context	Loaded handle	AES - DECKTS - Unwrap KB KDF MAC	Unauthenticated - contextEncKey: G,E,Z - objSeed: W - objSens: W - objPub: W - objAuth: W - nullProof: E - phProof: E - ehProof: E - shProof: E - sesHmacKey: W - seqAuth: W - contextKey: E
TPM2_ContextSave	Saves a session context, object context, or	Approved	Saved handle	Context	AES - ENC	Unauthenticated -

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	sequence object context outside the TPM				KTS-Wrap KBKDFMAC	contextEncKey: G,E,Z - objSeed: R - objSens: R - objPub: R - objAuth: R - nullProof: E - phProof: E - ehProof: E - shProof: E - sesHmacKey: R - seqAuth: R - contextKey: E
TPM2_Create (I/E/D)	Object creation	Approved	Parent object handle, Object sensitive part,	Object private part (encrypted), Object public part, Creation	KeyGen AES-ENC SigG	User (U) - objSeed: G,R,E - objSym

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			Object public template, Creation data, List of PCR	data, Digest of creation data, Ticket to be used by TPM2_CertifyCreation()	en SigVer DRBG KTS - Unwrap KTS - Wrap KB KDF MAC SHA SKG	Key: G,E,Z - objHmacKey: G,E,Z - objSens: G,R,E - objPub: G,R,E - drbgState: W,E - objAuth: W,R - nullProof: E - phProof: E - ehProof: E - shProof: E - eccParams256: E - eccParams384: E
TPM2_CreateLoaded (I/E/D)	Creates an object and loads it in the TPM	Approved	Parent object handle, Object	Object private part (encrypted), Object	KeyGen KeyVer	Crypto officer (CO) -

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			sensitive part, Object public template	public part, Creation object name	AES - ENC SigGen SigVer DRBG KB KDF MAC SHA SKG	objSeed: G,R,E - objSymKey: G,E - objHmacKey: G,E - objSens: G,R,E - objPub: G,R,E - tdrbgState: G,W,E - drbgState: W,E - objAuth: W,R - nullSeed: E - phSeed: G,E - ehSeed: G,E - shSeed: G,E - nullProof: E - phProof: E - ehProof: E

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- shProof: E - ekRsa: E - ekEcc: E - shProofForReseed: G,E - eccParams256: E - eccParams384: E User (U) - objSeed: G,R,E - objSymKey: G,E - objHmacKey: G,E - objSens: G,R,E - objPub: G,R,E - tdrbgState: G,W,E - drbgState: W,E -

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						objAuth: R,W - nullSeed: E - phSeed: E - shSeed: E - nullProof: E - phProof: E - ehProof: E - shProofForReseed: G,E - ekRsa: E - ekEcc: E - eccParams256: E - eccParams384: E - ehSeed: E - shProof: E

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_CreatePrimary (I/E/D)	Creates a Primary Object under one of the Primary Seeds or a Temporary Object under TPM_RH_NULL	Approved	Primary handle, Key sensitive data, Key public template, Creation data, Creation PCR	Object handle, Object Public part, Creation data, Digest of creation data, Creation ticket, Name of the object	KeyGen KeyVer SigGen SigVer DRBG KBKDF MAC SHA SKG	Crypto officer (CO) - objSeed: G,E,Z - objSymKey: G,E,Z - objHmacKey: G,E,Z - objSens: G,E,Z - objPub: G,R,E,Z - tdrbgState: G,W,E,Z - drbgState: W,E - objAuth: W - nullSeed: E - phSeed: G,E - ehSeed: G,E - shSeed: G,E - nullProof

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						: E - phProof: E - ehProof: E - shProof: E - ekRsa: E - ekEcc: E - shProofForReseed: G,E - eccParams256: E - eccParams384: E
TPM2_DictionaryAttackLockReset (I)	Cancels the effect of a TPM lockout due to several successive authorization failures	Non-security relevant	Authorization handle	None	None	Crypto officer (CO)
TPM2_DictionaryAttackParameters (I)	Changes the lockout parameters	Non-security relevant	Authorization handle, newMaxTries, newRecoveryTime, and	None	None	Crypto officer (CO)

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			lockoutRecovery values			
TPM2_Duplicate (I/E/D)	Duplicates a loaded object so that it may be used in a different hierarchy	Approved	Handle of the loaded object to duplicate, Handle of the new parent, Optional symmetric encryption key	Encryption key for inner wrapper, Duplicated object private part (encrypted), Encrypted seed	AES - ENCDRBG KTS - Encap KAS KTS - Wrap KB KDF MAC SHA SKG	User (U) - dupSeed: G,R,E,Z - objSeed: R - dupOutSymKey: G,E,Z - dupInSymKey: G,R,W,E,Z - dupOutHmacKey: G,E,Z - objSens: R - objAuth: R - drbgState: W,E - objPub: E - eccParams256: E - eccParams384:

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						E - intSharedSecret: G,E,Z - ephSensEccKey: G,E,Z - ephPubEccKey: G,R,E,Z
TPM2_ECC_Parameters (I)	Returns the parameters of an ECC curve identified by its TCG-assigned curveID	Non-security relevant	ID of an ECC curve	Curve parameters	None	Unauthenticated
TPM2_ECDH_KeyGen (I/E/D)	Shared secret value computation using ECDH	Approved	ECC key public part handle	Shared secret, Ephemeral public key	KAS-KeyGen KAS	Unauthenticated - ephSensEccKey: G,E,Z - ephPubEccKey: G,R,Z - drbgState: W,E - objPub: E - eccParams256: E - eccParams384:

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						E - extSharedSecret: G,R,Z
TPM2_ECDH_ZGen (I/E/D)	Shared secret value recovery using ECDH	Approved	Handle of a loaded ECC key, Ephemeral public key	Recovered shared secret	KAS	User (U) - ephPubEccKey: W,E,Z - objSens: E - eccParams256: E - eccParams384: E - extSharedSecret: G,R,Z
TPM2_EncryptDecrypt (I/E)	Symmetric encryption or decryption	Approved	Symmetric key handle, Decryption or encryption indicator, Encryption Mode, Data to be encrypted or decrypted,	Encrypted or decrypted data, Chaining value	AES - ENC AES - DEC	User (U) - objSens: E

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			Initialization Vector per NIST SP 800-38A or chaining value			
TPM2_EncryptDecrypt2 (I/E/D)	Symmetric encryption or decryption	Approved	Symmetric key handle, Decryption or encryption indicator, Encryption Mode, Data to be encrypted or decrypted, Initialization Vector per NIST SP 800-38A or chaining value	Encrypted or decrypted data, Chaining value	AES - ENC AES - DEC	User (U) - objSens: E
TPM2_EventSequenceComplete (I/D)	Adds last part of data to a hash or HMAC sequence and returns the result in a digest list	Approved	Handle of PCR to extend Sequence handle, Data to hash/HMAC	List of digests computed for the PCR	MAC SHA	User (U) - objSens: E - seqAuth: Z

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_EvictControl (I)	Allows certain Transient Objects to be made persistent or a persistent object to be evicted	Approved	Authorization handle, Loaded object handle, Persistent handle	None	None	Crypto officer (CO) - objSeed: W,Z - objSens: W,Z - objPub: W,Z - objAuth: W,Z - sesHmacKey: W - sesSymKey: W
TPM2_FlushContext	Causes all context associated with a loaded object, sequence object, or session to be removed from TPM memory	Approved	Flush handle	None	None	Unauthenticated - objSeed: Z - objSens: Z - objPub: Z - objAuth: Z
TPM2_GetCapability (I) - Show Version	Returns various information regarding the TPM and its current state	Non-security relevant	Capability, property, property count	More data availability indicator, Capability data	None	Unauthenticated

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_GetCommandAuditDigest (I/E/D)	Returns the current value of the command audit digest, a digest of the commands being audited, and the audit hash algorithm	Approved	Handle of a privacy administrator, Handle of a signing key, Qualifying data, Signature scheme	Audit information, Signature over the quoted information	SigGen DRBG KBKDF MAC SHA	Crypto officer (CO) - drbgState: W,E - objSens: E - shProof: E
TPM2_GetRandom (I/E)	Outputs random bytes from a DRBG	Approved	Number of random bytes to generate	Output random bytes	DRBG	Unauthenticated - drbgState: W,E
TPM2_GetSessionAuditDigest (I/E/D)	Returns a digital signature of the audit session digest	Approved	Handle of a privacy administrator, Handle of a signing key, Handle of an audit session, Qualifying data, Signature scheme	Audit information, Signature over the quoted information	SigGen KBKDF DRBG MAC SHA	Crypto officer (CO) - objSens: E - shProof: E - drbgState: W,E
TPM2_GetTestResult (I) - Show Status	Get self-tests result	Non-security relevant	None	Self-tests status	None	Unauthenticated

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_GetTime (I/E/D)	Returns the current values of Time and Clock	Approved	Handle of a privacy administrator, Handle of a signing key, Qualifying data, Signature scheme	Attestation data, Signature over the attestation data	SigGen KB KDF DRBG MAC SHA	Crypto officer (CO) - drbgState: W,E - objSens: E - shProof: E
TPM2_HMAC (I/E/D)	Performs a HMAC operation on data	Approved	Symmetric signing key handle, Data to HMAC Hash algorithm	HMAC	MAC	User (U) - objSens: E
TPM2_HMAC_Start (I/D)	Starts an HMAC sequence	Approved	Handle of an HMAC key, Authorization value for sequence Hash algorithm	Sequence handle	MAC	User (U) - seqAuth: W - objSens: E
TPM2_Hash (I/E/D)	Performs a hash operation on data	Approved	Data to hash, Hash algorithm Hierarch	Digest Ticket linked to the input hierarchy	MAC SHA	Unauthenticated - nullProof: E - phProof:

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			y to use for ticket			E - ehProof: E - shProof: E
TPM2_HashSequenceStart (I/D)	Starts a hash or an event sequence	Approved	Authorization value for sequence Hash algorithm	Sequence handle	SHA	Unauthenticated - seqAuth: W
TPM2_HierarchyChangeAuth (I/D)	Changes the authValue of hierarchies	Non-security relevant	Authorization handle, New authorization value	None	None	Crypto officer (CO) - lockoutAuth: W - endorsementAuth: W - ownerAuth: W - platformAuth: W
TPM2_HierarchyControl (I)	Enables and disables use of a hierarchy and its associated NV storage	Non-security relevant	Primary handle, Hierarchy to enable or disable, Enable or disable indicator	None	None	Crypto officer (CO)

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_Import (I/E/D)	Allows an object to be encrypted using the symmetric encryption values of a Storage Key	Approved	Handle of the new parent, Duplicated object private part (encrypted), Object public part, Encrypted seed, Encryption key for inner wrapper	Object private part (encrypted)	AES - ENC AES - DEC KAS KTS - Decap KTS - Wrap KTS - Unwrap KB KDF MAC SHA	User (U) - objSens: R,W,E,Z - objSeed: R,W,Z - objPub: W,E,Z - dupOutSymKey: W,E,Z - objAuth: R,W,Z - drbgState: W,E - dupSeed: E,W,Z - dupInSymKey: E,W,Z - dupOutHmacKey: W,E,Z - eccParams256: E - eccParams384: E - intSharedSecret:

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						G,E,Z - extPubEccKey: W,E,Z
TPM2_IncrementalSelfTest (I)	Incremental self-tests execution	Approved	List of tests to pass	List of remaining tests		Unauthenticated - eccParams256: E
TPM2_Init	Reboot or power-on of the TPM.	Non-security relevant	None	None	None	Unauthenticated - nullSeed: Z - nullProof: Z - platformAuth: Z - objSeed: Z - objAuth: Z - objSens: Z - objPub: Z - sesSalt: Z - sesHmacKey: Z - sesSymKey: Z

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- contextKey: Z - drbgSeed: Z - objSymKey: Z - objHmacKey: Z - contextEncKey: Z - dupSeed: Z - dupInSymKey: Z - dupOutSymKey: Z - dupOutHmacKey: Z - creSeed: Z - creSymKey: Z - creHmacKey: Z - ephSensEccKey:

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						Z - ephPubEccKey: Z - seqAuth: Z - tdrbgState: Z - fuSymKey: Z - drbgState: Z
TPM2_Load (I/E/D)	Object loading	Approved	Parent object handle, Object private part (encrypted), Object public part	Name of the loaded object	Key Ver AES - DEC KTS - Unwrap KB KDF MAC SHA	User (U) - objSymKey: G,W,E,Z - objHmacKey: G,W,E,Z - objSens: W,E - objPub: W,E - objSeed: W,E - objAuth: W - eccParams256: E -

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						eccParams384: E
TPM2_LoadExternal (I/E/D)	External object loading	Approved	Object public part, Hierarchy	Name of the loaded object	Key Val	Unauthenticated - objPub: W,E - objSens: W,E - objAuth: W - objSeed: W - eccParams256: E - eccParams384: E
TPM2_MakeCredential (I/E/D)	Allows the TPM to perform the actions required of a Certificate Authority (CA) in creating a TPM2B_ID_OBJECT containing an activation credential	Approved	Handle of a loaded public key, Credential information, Name of the object with credentials	Encrypted credential, Encrypted seed	AES - ENC KAS KAS - Key Gen KTS - Encap KTS - Wrap KB	Unauthenticated - creSeed: G,R,E,Z - creSymKey: G,E,Z - creHmacKey: G,E,Z - objPub: E - eccPara

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
					KDF MAC SHA	ms256: E - eccParams384: E - ephSensEccKey: G,E,Z - ephPubEccKey: G,R,Z - intSharedSecret: G,E,Z
TPM2_NV_Certify (I/E/D)	Certifies the contents of an NV Index or portion of an NV Index	Approved	Handle of signing key, Authorization handle NV index, Qualifying data Scheme, Size and offset in NV area	Structure that was signed, Signature	SigGen KB KDF MAC SHA	User (U) - objSens: E - shProof: E
TPM2_NV_ChangeAuth (I/D)	Allows the authValue of an NV Index to be changed	Approved	NV index, New authorization value	None	None	User (U) - nvAuth: W

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_NV_DefineSpace (I/D)	Defines the attributes of an NV Index and causes the TPM to reserve space to hold the data associated with the NV Index	Approved	Authorization handle, NV authorization value, NV public parameters	None	None	Crypto officer (CO) - nvAuth: W
TPM2_NV_Extend (I/D)	Extends a value to an area in NV memory that was previously defined by TPM2_NV_DefineSpace()	Approved	Authorization handle, NV index to extend, Data to extend	None	SHA	User (U)
TPM2_NV_GlobalWriteLock (I)	Sets TPMA_NV_WRITELOCKED for all indexes that have their TPMA_NV_GLOBALLOCK attribute SET	Non-security relevant	Authorization handle	None	None	Crypto officer (CO)
TPM2_NV_Increment (I)	Increments the value in an NV Index that has the TPM_NT_COUNTER attribute	Non-security relevant	Authorization handle, NV index to increment	None	None	User (U)
TPM2_NV_Read (I/E)	Reads a value from an area in NV memory previously defined by TPM2_NV_DefineSpace()	Non-security relevant	Authorization handle, NV index to be read, Size and	Data read	None	User (U)

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			offset in NV area			
TPM2_NV_ReadLock (I)	Prevents further reads of the NV Index until the next TPM2_Startup (TPM_SU_CLEAR) if TPMA_NV_READ_STCLEAR is SET	Non-security relevant	Authorization handle, NV index to be locked	None	None	User (U)
TPM2_NV_ReadPublic (I/E)	Reads the public area and Name of an NV Index	Approved	NV index	NV index public area, Name of the NV index	SHA	Unauthenticated
TPM2_NV_SetBits (I)	Sets bits in an NV Index that was created as a bit field	Non-security relevant	Authorization handle, NV index to extend, Data to OR with NV content	None	None	User (U)
TPM2_NV_UndefineSpace (I)	Removes an Index from the TPM	Approved	Authorization handle, NV index to delete	None	None	Crypto officer (CO) - nvAuth: Z
TPM2_NV_UndefineSpaceSpecial (I)	Removal of a platform-created NV Index that has TPMA_NV_POLICY_DELETE SET	Approved	Platform authorization handle, NV index to delete	None	None	Crypto officer (CO) - nvAuth: Z

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_NV_Write (I/D)	Writes a value to an area in NV memory that was previously defined by TPM2_NV_DefineSpace()	Non-security relevant	Authorization handle, NV index to write, Data to write, Offset in the NV index area	None	None	User (U)
TPM2_NV_WriteLock (I)	Inhibits further writes of the NV Index if the TPMA_NV_WRITEDEFINE or TPMA_NV_WRITE_STCLEAR attributes of an NV location are SET	Non-security relevant	Authorization handle, NV index	None	None	User (U)
TPM2_ObjectChangeAuth (I/E/D)	Changes the authorization secret for a TPM-resident object	Approved	Handle of an object, Handle of the parent of the object, New authorization value	Object private part	AES - ENC KB KDF MA C SHA	User (U) - objSeed: R,E - objSens: R - drbgState: W,E - objAuth: R,W - objSymKey: G,E -

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						objHmacKey: G,E
TPM2_PCR_Allocate (I)	Sets the desired PCR allocation of PCR and algorithms	Non-security relevant	Selection of PCR to allocate	PCR allocation information	None	Crypto officer (CO)
TPM2_PCR_Event (I/D)	Updates the indicated PCR and reports list of digests	Approved	PCR handle, Event data	Digests	SHA	Unauthenticated
TPM2_PCR_Extend (I)	Updates the indicated PCR	Approved	PCR handle, List of digests used to extend PCRs	None	SHA	Unauthenticated
TPM2_PCR_Read (I)	Returns the values of all PCR specified in pcrSelectionIn	Non-security relevant	Selection of PCR to read	PCR information		Unauthenticated
TPM2_PCR_Reset (I)	Sets the PCR in all banks to zero	Non-security relevant	PCR to reset	None		Unauthenticated
TPM2_PP_Commands	Determines which commands require assertion of Physical Presence	Non-security relevant	Authorization handle, List of commands to add and list of comman	None	None	Crypto officer (CO)

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			d to remove			
TPM2_PolicyAuthValue (I)	Allows a policy to be bound to the authorization value of the authorized entity	Approved	Policy session handle	None	SHA	Unauthenticated
TPM2_PolicyAuthorize (I/D)	Check a ticket issued from the signature verification of a new policy so that it may be used in an existing policy	Approved	Policy session handle, Digest of the policy being approved, Policy qualifier, Key name, Ticket	None	MAC SHA	Unauthenticated - nullProof: E - phProof: E - ehProof: E - shProof: E
TPM2_PolicyAuthorizeNV (I)	Provides a capability that is the equivalent of a revocable policy	Approved	Source handle for authorization, NV index to read, Policy session handle	None	SHA	User (U)
TPM2_PolicyCommandCode (I)	Limits policy to a specific command code	Approved	Policy session handle, Command code	None	SHA	Unauthenticated
TPM2_PolicyCounterTimer (I/D)	Causes conditional gating of a	Approved	Policy session handle,	None	SHA	Unauthenticated

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	policy based on the contents of the TPMS_TIME_INFO structure		Operand, offset, operation			
TPM2_PolicyCpHash (I/D)	Allows a policy to be bound to a specific command and command parameters	Approved	Policy session handle, Digest to add to policy	None	SHA	Unauthenticated
TPM2_PolicyDuplicationSelect (I/D)	Allows qualification of duplication to allow duplication to a selected new parent	Approved	Policy session handle, Object name to be duplicated, New Parent name, Object name inclusion indicator	None	SHA	Unauthenticated
TPM2_PolicyGetDigest (I/E)	Returns the current policyDigest of a policy session	Non-security relevant	Policy session handle	Policy digest	None	Unauthenticated
TPM2_PolicyLocality (I)	Indicates that the policy will be limited to a specific locality	Approved	Policy session handle, Locality	None	SHA	Unauthenticated
TPM2_PolicyNV (I/D)	Causes conditional gating of a policy based on the contents of an NV Index	Approved	Authorization handle, NV index handle,	None	SHA	User (U)

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			Policy session handle, Operand, offset, operation			
TPM2_PolicyNameHash (I/D)	Allows a policy to be bound to a specific set of TPM entities without being bound to the parameters of the command	Approved	Policy session handle, Digest to add to policy	None	SHA	Unauthenticated
TPM2_PolicyNvWritten (I)	Allows a policy to be bound to the TPMA_NV_WRITTEN attributes	Approved	Policy session handle, NV index written indicator	None	SHA	Unauthenticated
TPM2_PolicyOR (I)	Allows options in authorizations without requiring that the TPM evaluate all the options	Approved	Policy session handle, List of digests	None	SHA	Unauthenticated
TPM2_PolicyPCR (I/D)	Causes conditional gating of a policy based on PCR	Approved	Policy session handle, Expected digest value PCR selection	None	SHA	Unauthenticated
TPM2_PolicyPassword (I)	Allows a policy to be bound to the authorization	Approved	Policy session handle	None	SHA	Unauthenticated

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	value of the authorized object					
TPM2_PolicyPhysicalPresence (I)	Physical presence will need to be asserted at the time the authorization is performed	Approved	Policy session handle	None	SHA	Unauthenticated
TPM2_PolicyRestart (I)	Policy session restart	Non-security relevant	Session handle	None	None	Unauthenticated
TPM2_PolicySecret (I/E/D)	Includes a secret-based authorization to a policy	Approved	Authorization object handle, Policy session handle, Nonce TPM Digest, Expiration of authorization, Policy reference value	Policy timeout, Policy ticket	MAC SHA	User (U) - nullProof: E - phProof: E - ehProof: E - shProof: E
TPM2_PolicySigned (I/E/D)	Includes a signed authorization in a policy	Approved	Signature key handle, Policy session handle, Nonce TPM	Policy timeout, Policy ticket	SigVer MAC SHA	Unauthenticated - objPub: E - nullProof: E -

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			Digest, Signature, Expiration of authorization, Policy reference value			phProof: E - ehProof: E - shProof: E
TPM2_PolicyTemplate (I/D)	Allows a policy to be bound to a specific creation template	Approved	Policy session handle, Digest to add to policy	None	SHA	Unauthenticated
TPM2_PolicyTicket (I/D)	Includes a ticket in a policy	Approved	Policy session handle, Nonce TPM Digest, Expiration of authorization, Policy reference value, Authorization object name, Ticket	None	MAC SHA	Unauthenticated - nullProof: E - phProof: E - ehProof: E - shProof: E
TPM2_Quote (I/E/D)	Quotes PCR values	Approved	Handle of a signing key, Qualifying data, Selection	Quoted information, Signature over the quoted information	SigGen DRBG KBKDF MA	User (U) - drbgState: W,E - objSens: E

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			of PCRs Signature scheme		C SHA	- shProof: E
TPM2_RSA_Decrypt (I/E/D)	Performs RSA decapsulation for the purpose of key transport only	Approved	RSA private key handle, Ciphertext to decrypt RSA scheme to use	Decrypted output	KTS - Decap	User (U) - objSens: E
TPM2_RSA_Encrypt (I/E/D)	Performs RSA encapsulation for the purpose of key transport only	Approved	RSA public key handle, Message to encrypt RSA scheme to use	Encrypted output	KTS - Encap	Unauthenticated - objPub: E
TPM2_ReadClock (I)	Reads the current TPMS_TIME_INFO structure	Non-security relevant	None	Current time	None	Unauthenticated
TPM2_ReadPublic (I)	Read public part of a loaded object	Approved	Handle of an object	Object public part, Object name, Object qualified name	None	Unauthenticated - objPub: R
TPM2_Rewrap (I/E/D)	Rewraps a duplicated object with a new parent key	Approved	Handle of the old parent, Handle of the	Duplicated object private part (encrypted),	AES - ENC AES -	User (U) - dupOutSymKey: G,E,Z

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
			new parent, Duplicated object private part (encrypted), Name of the object being rewrapped, Encrypted seed	Encrypted seed	DEC KAS KAS-KeyGen KTS-Decap KTS-Encap KTS-Wrap KTS-Unwrap KB KDF MAC SHA	- dupOutHmacKey: G,E,Z - objSens: R,W,E - dupSeed: R,W,E,Z - objSeed: R,W - dupInSymKey: W,Z - drbgState: W,E - objPub: E - objAuth: R,W - eccParams256: E - eccParams384: E - intSharedSecret: G,E,Z - ephSensEccKey: G,E,Z

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- ephPubEccKey: G,R,E,Z
TPM2_SelfTest (I)	Self-tests execution	Approved	Full or background self-tests	Self-test result if full self-tests required		Unauthenticated - eccParams256: E
TPM2_SequenceComplete (I/E/D)	Adds last part of data to a hash or HMAC sequence and returns the result	Approved	Sequence handle, Data to hash/HMAC, Hierarchy for ticket	HMAC or digest, Ticket linked to the input hierarchy	MAC SHA	User (U) - nullProof: E - phProof: E - ehProof: E - shProof: E - objSens: E - seqAuth: Z
TPM2_SequenceUpdate (I/D)	Adds data to a hash or HMAC sequence	Approved	Sequence handle, Data to hash/HMAC	None	MAC SHA	User (U) - objSens: E
TPM2_SetCapability (I/D)	Set specific data in the TPM, such as TPM configurations, which may change the	Non-security relevant	Capability data	None	None	Crypto officer (CO)

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	TPM's function and behavior					
TPM2_SetCommandCodeAuditStatus (I)	Changes the audit status of a command or to set the hash algorithm used for the audit digest	Non-security relevant	Authorization handle, Hash algorithm	None	None	Crypto officer (CO)
TPM2_SetPrimaryPolicy (I/D)	Sets the authorization policy for a hierarchy	Non-security relevant	Primary handle, Policy digest, Hash algorithm	None	None	Crypto officer (CO)
TPM2_Shutdown (I)	Prepare the TPM for a power cycle.	Non-security relevant	Shutdown type	None	None	Unauthenticated
TPM2_Sign (I/D)	Causes the TPM to sign an externally provided hash with the specified symmetric or asymmetric signing key	Approved	Handle of a signing key, Digest to be signed, Scheme, Proof ticket for digest	Signature over the digest	SigGen DRBG MAC SHA	User (U) - objSens: E - nullProof: E - phProof: E - ehProof: E - shProof: E - drbgState: W,E

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_StartAuthSession (I/E/D)	Session command	Approved	Decryption key handle, Binding entity handle, Encrypted salt, Nonce caller, Session Type (HMAC or Policy)	Nonce TPM	KAS KTS - Decap KB KDF	Unauthenticated - sesHmacKey: G,W - sesSalt: G,W,E,Z - objSens: E - objAuth: E - nvAuth: E - platformAuth: E - endorsementAuth: E - ownerAuth: E - lockoutAuth: E - seqAuth: E - eccParams256: E - eccParams384: E

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						- extPubEccKey: W,E,Z - intSharedSecret: G,E,Z
TPM2_Startup	Set-up the TPM after a power cycle.	Approved	Startup type	None	ENT - ESV DRBG	Unauthenticated - phSeed: G - ehSeed: G - shSeed: G - phProof: G - ehProof: G - shProof: G - contextKey: G - drbgSeed: G - drbgState: G - nullSeed: G - nullProof: G

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_StirRandom (I/D)	Reseed the state of a DRBG	Approved	Additional information	None	DRBG ENT - ESV	Unauthenticated - drbgSeed: W,E,Z - drbgState: W,E
TPM2_TestParms (I)	Checks if specific combinations of algorithm parameters are supported	Non-security relevant	Algorithm parameters	None	None	Unauthenticated
TPM2_Unseal (I/E/D)	Returns the data in a loaded Sealed Data Object	Approved	Handle of a loaded data object	Unsealed data	None	User (U)
TPM2_VendorCmdChangeObjectDeletionAuth (I)	Modifies deletion authorization for an object	Non-security relevant	Authorization handle, Platform authorization use indicator	None	None	Crypto officer (CO)
TPM2_VendorCmdFieldUpgradeData (I)	Conveys firmware in a field upgrade session	Approved	Field upgrade data blob	Completion indicator	AES - DEC SHA	Unauthenticated - fuSymKey: E,Z
TPM2_VendorCmdFieldUpgradeStart (I)	Initiates a field upgrade session	Approved	Authorization handle, Field upgrade metadata blob	None	SigVer KB KDF SHA	Crypto officer (CO) - fuSigECCKey: E - fuSymKey: G -

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
						fuSymSeed: E
TPM2_VendorCmdGPIOConfig (I)	Configures GPIO	Non-security relevant	Authorization handle GPIO configuration	None	None	Unauthenticated
TPM2_VendorCmdGetRandom2 (I/E)	Get random value from DRBG	Approved	Number of bytes to generate	Random value	DRBG	Unauthenticated - drbgState: W,E
TPM2_VendorCmdGetRandom800_90B (I/E)	Get random value from ESV Cert. #E41	Approved	Number of bytes to generate	Random value	ENT-ESV	Unauthenticated
TPM2_VendorCmdRestoreEK (I)	Restore EK RSA or EK ECC in case of deletion by TPM2_ChangeEPS	Approved	Authorization handle	None	None	Crypto officer (CO) - ekRsa: W - ekEcc: W
TPM2_VendorCmdSetCommandSet (I)	Activates and locks commands	Non-security relevant	Authorization handle, Command code, Activation and lock indicators	None	None	Crypto officer (CO)
TPM2_VendorCmdSetCommandSetLock (I)	Prevents locking commands	Non-security relevant	Authorization handle	None	None	Crypto officer (CO)

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
TPM2_VendorCmdSetMode (I)	Sets the low power mode	Non-security relevant	Authorization handle, Low power configuration structure	None	None	Crypto officer (CO)
TPM2_VendorCmdZeroizeEK (I)	Zeroize EK RSA and EK ECC	Approved	Authorization handle	None	None	Crypto officer (CO) - ekRsa: Z - ekEcc: Z
TPM2_VerifySignature (I/D)	Uses loaded keys to validate a signature on a message with the message digest passed to the TPM	Approved	Handle of a public key, Digest of a message, Signature to be tested	Validation ticket	SigVer MAC	Unauthenticated - objPub: E - nullProof: E - phProof: E - ehProof: E - shProof: E
_TPM_Hash_Data	Indicates to the TPM interface data to be included in the H-CRTM measurement sequence	Approved	Data	None	SHA	Unauthenticated
_TPM_Hash_End	Indicates to the TPM interface the end of the	Approved	None	None	SHA	Unauthenticated

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Indicator	Inputs	Outputs	Security Functions	SSP Access
	H-CRTM measurement sequence					
_TPM_Hash_Start	Indicates to the TPM interface the start of an H-CRTM measurement sequence	Approved	None	None	SHA	Unauthenticated

Table 24: Approved Services

The integrity mechanism provided by sessions is not directly callable from the security module external interfaces. Function is used (or might be used) by the services listed in this table. When a service is usable with a session, (I) is added next to the service name. When a service can additionally use the encryption mechanism of a session, (I/E) is added next to the service name.

The encryption mechanism provided by sessions is not directly callable from the security module external interfaces. Function is used (or might be used) by the services listed in this table. When a service is usable with a session, (I) is added next to the service name. When a service can additionally use the encryption mechanism of a session, (I/E) is added next to the service name.

Public Material – May be reproduced only in its original entirety (without revision).

4.4 Non-Approved Services

All approved services implemented by the Module are listed in the table below:

Name	Description	Algorithms	Role
TPM2_Create; TPM2_CreateLoaded; TPM2_Load; TPM2_LoadExternal	Creation or loading of an ECC key with a non-approved elliptic curve; Creation or loading of an ECC key for a non-approved key agreement usage; Creation or loading of an ECC signing key with an undetermined scheme (field inPublic.buffer.parameters.scheme.scheme = TPM_ALG_NULL);Creation or loading of an RSA decryption key with an undetermined scheme (field inPublic.buffer.parameters.scheme.scheme = TPM_ALG_NULL); Creation or loading of a 1024-bit RSA key	ECC BN P-256 (non-compliant) RSA (non-compliant)	User
TPM2_CreateLoaded	Derivation of an ECC key from a derivation parent key	ECC derived keys (non-compliant) KBKDF (non-compliant)	User
TPM2_Load; TPM2_LoadExternal	Loading of an ECC or RSA key (sensitive and public parts) in the NULL hierarchy	ECC BN P-256 (non-compliant) RSA (non-compliant)	User
TPM2_Duplicate; TPM2_Rewrap; TPM2_Import	Key transport with a 1024-bit RSA key Key agreement scheme with a non-approved ECC curve Key agreement scheme with an ECC key used in a non-approved key agreement usage	ECC BN P-256 (non-compliant) KAS (non-compliant) RSA (non-compliant)	User

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Algorithms	Role
TPM2_RSA_Encrypt; TPM2_RSA_Decrypt	Key transport with a non-approved scheme: * RSAES-PKCS1-v1_5 * RSA with no padding mode (null scheme) Key transport with an RSA decryption key: * Generated with an undetermined scheme (field inPublic.buffer.parameters.scheme.scheme = TPM_ALG_NULL) * Loaded in the NULL hierarchy	KTS-IFC (non-compliant) RSA with no padding mode (null scheme) (non-compliant) RSAES-PKCS1-v1_5 (non-compliant)	User
TPM2_ECDH_KeyGen	Use of a non-approved elliptic curve: * ECC key with curve BN P-256 Use of an ECC key for a non-approved key agreement usage: * ECC key with curve Curve448	ECC BN P-256 (non-compliant)	N/A
TPM2_ECDH_ZGen	Use of an ECC key: * Generated on curve BN P-256 * For a non-approved key agreement usage * Derived from a derivation parent key * Loaded in the NULL hierarchy	ECC BN P-256 (non-compliant) KBKDF (non-compliant)	User
TPM2_ZGen_2Phase	This command is only usable jointly with TPM2_EC_Ephemeral service that is non approved as using key derivation to generate ECC keys	ECC derived keys (non-compliant) KBKDF (non-compliant)	User
TPM2_HMAC	HMAC generation with a key length < 112 bits	HMAC (non-compliant)	User
TPM2_HMAC_Start; TPM2_SequenceUpdate; TPM2_SequenceComplete	HMAC generation with a key length < 112 bits	HMAC (non-compliant)	User
TPM2_Certify; TPM2_CertifyCreation; TPM2_Quote;	Digital signature with a non-approved signature scheme: * ECC signature with ECDAA signature scheme * ECC	ECC BN P-256 (non-	User/CO

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Algorithms	Role
TPM2_GetSessionAuditDigest; TPM2_GetCommandAuditDigest; TPM2_GetTime; TPM2_CertifyX509	signature with ECSchnorr signature scheme * RSA signature with key length of 1024 bits * ECC or RSA signature key using SHA-1 as digest method * ECC signature with curve BN P-256 Digital signature with an ECC signing key generated with an undetermined scheme (field inPublic.buffer.parameters.scheme.scheme = TPM_ALG_NULL); Digital signature with an ECC signing derived from a derivation parent key; Digital signature with an ECC or RSA key loaded in the NULL hierarchy	compliant) ECDAA (non-compliant) ECDSA (non-compliant) ECSchnorr (non-compliant) RSA (non-compliant) SHA-1 (non-compliant)	
TPM2_Commit	Generation of an ECC key through non-Approved key derivation method	KBKDF (non-compliant)	User
TPM2_EC_Ephemeral	Generation of an ECC key through non-Approved key derivation method	KBKDF (non-compliant)	User
TPM2_VerifySignature	Digital signature verification with a non-approved signature scheme or a non-approved curve: * ECDAA signature scheme * ECSchnorr signature scheme * ECC signature with curve BN P-256	ECC BN P-256 (non-compliant) ECDAA (non-compliant) ECSchnorr (non-compliant)	NA
TPM2_Sign	Digital signature generation with a non-approved signature scheme: * ECC signature with ECDAA signature scheme * ECC signature with ECSchnorr signature scheme * RSA signature with key length of 1024 bits * ECC or RSA signature key using SHA-1 as digest method * ECC signature with curve BN P-256; Digital signature with an ECC signing key generated with an undetermined scheme (field inPublic.buffer.parameters.scheme.scheme =	ECC BN P-256 (non-compliant) ECDAA (non-compliant) ECDSA (non-compliant) ECSchnorr	User

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Algorithms	Role
	TPM_ALG_NULL); Digital signature with an ECC signing derived from a derivation parent key; Digital signature with an ECC or RSA key loaded in the NULL hierarchy	(non-compliant) RSA (non-compliant) SHA-1 (non-compliant)	
TPM2_PolicySigned	Digital signature verification with a non-approved signature scheme or a non-approved curve: * ECDAA signature scheme * ECSchnorr signature scheme * ECC signature with curve BN P-256	ECC BN P-256 (non-compliant) ECDAA (non-compliant) ECSchnorr (non-compliant)	N/A
TPM2_CreatePrimary	Creation and loading of an ECC key with a non-approved elliptic curve: * ECC key with curve BN P-256 Use of an ECC key for a non-approved key agreement usage: * ECC key with curve Curve448 Creation and loading of an ECC signing key with an undetermined scheme (field inPublic.buffer.parameters.scheme.scheme = TPM_ALG_NULL) Creation and loading of an RSA decryption key with an undetermined scheme (field inPublic.buffer.parameters.scheme.scheme = TPM_ALG_NULL)	ECC BN P-256 (non-compliant)	CO
TPM2_NV_Certify	Digital signature with a non-approved signature scheme: * ECC signature with ECDAA signature scheme * ECC signature with ECSchnorr signature scheme * RSA signature with key length of 1024 bits * ECC or RSA signature key using SHA-1 as digest method * ECC signature with curve BN P-256 Digital signature with an ECC signing key generated with an undetermined scheme (field inPublic.buffer.parameters.scheme.scheme =	ECC BN P-256 (non-compliant) ECDAA (non-compliant) ECDSA (non-compliant) ECSchnorr	User

Name	Description	Algorithms	Role
	TPM_ALG_NULL) Digital signature with an ECC signing derived from a derivation parent key Digital signature with an ECC or RSA key loaded in the NULL hierarchy	(non-compliant) RSA (non-compliant) SHA-1 (non-compliant)	

Table 25: Non-Approved Services

Public Material – May be reproduced only in its original entirety (without revision).

4.5 External Software/Firmware Loaded

Loading of firmware on the Module can be achieved by using two services:

- TPM2_VendorCmdFieldUpgradeStart that performs the software/firmware load test detailed in the self-test section of this document to determine if the authorizations to start a loading session are granted
- TPM2_VendorCmdFieldUpgradeData that transports the protected (confidentiality and integrity) parts of the firmware. Several commands are necessary to transport the full firmware.

Data outputs are inhibited until the loading session has completed successfully. Execution of the successfully loaded firmware is only effective after the next reset of the security module.

New firmware versions must be validated through the FIPS 140-3 validation process. Any other firmware loaded into this module is out of the scope of this validation and requires a separate FIPS 140-3 validation.

The core memory loader (CML) represented in Figure 9 is non-modifiable, only the TPM instances are modifiable by using an authenticated firmware upgrade mechanism. The Module contains two instances of the firmware but only one instance is executed after a boot sequence.

Public Material – May be reproduced only in its original entirety (without revision).

5 Software/Firmware Security

5.1 Integrity Techniques

The Module is composed of the following firmware component(s):

- Core Memory Loader executable (aka CML)
- TPM application instance 1 executable
- TPM application instance 2 executable

The firmware integrity is verified by computing a CRC-16 [ISO 13239] over the active firmware and comparing it to a reference value. Firmware integrity is verified during the boot sequence before the execution of the code blocks (CML and TPM). If a failure is detected during the boot sequence, the TPM enters an infinite reset loop that can be exit only by a power-off/power-on sequence. If a failure is detected during the self-tests execution, the security Module enters failure mode.

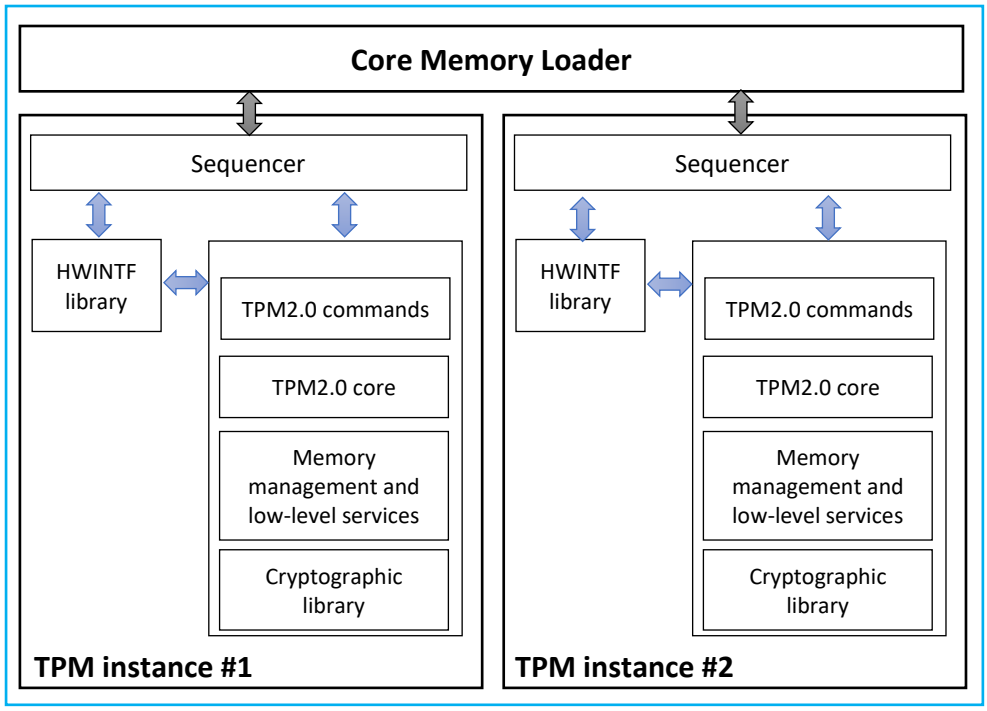


Figure 9 – Firmware block diagram

5.2 Initiate on Demand

The operator can initiate the integrity test on demand by using the TPM2_SelfTest command with the full parameter set to YES or by using the TPM2_IncrementalSelfTest command.

Public Material – May be reproduced only in its original entirety (without revision).

6 Operational Environment

6.1 Operational Environment Type and Requirements

Type of Operational Environment: Limited

The operational environment of the Module is “limited” because it allows loading authenticated firmware that meets all applicable requirements of [140-3] standard.

Data outputs are inhibited until the loading session has completed successfully. Execution of the successfully loaded FW is only effective after the next reset of the security module.

New firmware versions must be validated through the FIPS 140-3 validation process. Any other firmware loaded into this module is out of the scope of this validation and requires a separate FIPS 140-3 validation.

The core memory loader (CML) represented in Figure 9 is non-modifiable, only the TPM instances are modifiable by using an authenticated firmware upgrade mechanism.

The security module contains two instances of the FW but only one FW instance is executed after a boot sequence.

Public Material – May be reproduced only in its original entirety (without revision).

7 Physical Security

The security module is production grade and meets the Physical Security protection requirements for single-chip module at FIPS 140-3 Security Level 3.

7.1 Mechanisms and Actions Required

Zeroisation

Zeroisation of CSPs can be triggered by specific services as detailed in Section 9.3. It occurs in a sufficiently small time-period to prevent the recovery of the sensitive data between start of zeroisation and the zeroisation completeness.

Physical security mechanisms

The security module is encapsulated in a hard opaque package to prevent direct observation of internal security components. It implements additional security mechanisms:

- An active metal shield, located inside the package and covering the internal circuitry and the memory components. Cutting, removing, or modifying the shield layer will cause the security module to reset and enter a shutdown mode.
- An internal circuitry detecting environmental conditions outside the nominal operating range. Power supply voltage and temperature are continuously monitored. If conditions exist outside the range determined by the tamper detection circuitry, the security module resets and enters a failure mode. The module remains in failure mode as long as the environmental condition causing the tamper event persists.

Physical security inspection

Mechanism	Inspection Frequency	Inspection Guidance
Hard opaque package	Dependent on the security module integration environment varies from once per month to once per year	Visual inspection of the package to confirm that it has not been damaged by an external action

Table 26: Mechanisms and Actions Required

7.2 User Placed Tamper Seals

N/A

7.3 Filler Panels

N/A

7.4 Fault Induction Mitigation

N/A

Public Material – May be reproduced only in its original entirety (without revision).

7.5 EFP/EFT Information

EFT has been performed for all security module configurations. Low and high temperatures have been measured at a nominal voltage of 3.3V. Low and high voltage have been measured at ambient temperature (25°C).

The nominal operating ranges are:

- Between 1.62V and 3.8V for voltage
- Between -40°C and +125°C for temperature

Temp/Voltage Type	Temperature or Voltage	EFP or EFT	Result
LowTemperature	-60C (ST33KTPM2XSPI) / -70C (ST33KTPM2X) / -70C (ST33KTPM2A) / -70C (ST33KTPM2I)	EFT	Shutdown
HighTemperature	145C (ST33KTPM2XSPI) / 145C (ST33KTPM2X) / 145C (ST33KTPM2A) / 160C (ST33KTPM2I)	EFT	Shutdown
LowVoltage	1.5V (ST33KTPM2XSPI) / 1.4V (ST33KTPM2X) / 1.4V (ST33KTPM2A) / 1.4V (ST33KTPM2I)	EFT	Shutdown
HighVoltage	4.3V (ST33KTPM2XSPI) / 4.3V (ST33KTPM2X) / 4.3V (ST33KTPM2A) / 4.3V (ST33KTPM2I)	EFT	Shutdown

Table 27: EFP/EFT Information

7.6 Hardness Testing Temperature Ranges

Hardness testing was conducted at the temperature indicated in the table below.

Temperature Type	Temperature
LowTemperature	-65C
HighTemperature	150C

Table 28: Hardness Testing Temperatures

Public Material – May be reproduced only in its original entirety (without revision).

8 Non-Invasive Security

8.1 Mitigation Techniques

The Module does not claim support of non-invasive attack mitigation techniques referenced in [140F].

Public Material – May be reproduced only in its original entirety (without revision).

9 Sensitive Security Parameters Management

9.1 Storage Areas

Storage Area Name	Description	Persistence Type
Dynamic RAM	Volatile memory used to store SSPs between two consecutive resets or power-on/power-off sequence of the security module. SSPs don't persist after command execution. This area is marked as RAM on the HW block diagram.	Dynamic
Static RAM	Volatile memory used to store SSPs between two consecutive resets or power-on/power-off sequence of the security module. SSPs persist after command execution. This area is marked as RAM on the HW block diagram.	Static
NVRAM	Non-volatile memory (flash-based) used to store SSPs and make them persistent to a reset or a power-off/power-on sequence of the security module. This area is marked as flash memory on the HW block diagram.	Static

Table 29: Storage Areas

9.2 SSP Input-Output Methods

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Input plaintext to NVRAM	Outside of cryptographic boundary	NVRAM	Plaintext	Manual	Electronic	
Input protected to NVRAM	Outside of cryptographic boundary	NVRAM	Encrypted	Automated	Electronic	KTS-Unwrap
Input plaintext to RAM	Outside of cryptographic boundary	Static RAM	Plaintext	Manual	Electronic	
Input protected to RAM	Outside of cryptographic boundary	Static RAM	Encrypted	Automated	Electronic	KTS-Unwrap
Output plaintext from NVRAM	NVRAM	Outside of cryptographic boundary	Plaintext	Manual	Electronic	

Public Material – May be reproduced only in its original entirety (without revision).

Name	From	To	Format Type	Distribution Type	Entry Type	SFI or Algorithm
Output protected from NVRAM	NVRAM	Outside of cryptographic boundary	Encrypted	Automated	Electronic	KTS-Wrap
Output plaintext from RAM	Static RAM	Outside of cryptographic boundary	Plaintext	Manual	Electronic	
Output protected from RAM	Static RAM	Outside of cryptographic boundary	Encrypted	Automated	Electronic	KTS-Wrap
Input asym. encrypted to RAM	Outside of cryptographic boundary	Static RAM	Encrypted	Automated	Electronic	KTS-Decap
Output asym. encrypted from RAM	Static RAM	Outside of cryptographic boundary	Encrypted	Automated	Electronic	KTS-Encap

Table 30: SSP Input-Output Methods

Public Material – May be reproduced only in its original entirety (without revision).

9.3 SSP Zeroization Methods

Zeroization Method	Description	Rationale	Operator Initiation
TPM2_Init	Zeroization of all volatile SSPs. Explicit zeroization indicator provided by service completion status.	N/A	Activation of reset signal
TPM2_Clear	Zeroization of all contexts associated with an Owner. Explicit zeroization indicator provided by service completion status.	SSPs linked to an Owner must not persist if the Owner changes	Send TPM2_Clear command
TPM2_ChangePPS	Zeroize the platform primary seed and flush all transient and persistent objects in the Platform hierarchy. Explicit zeroization indicator provided by service completion status.	Platform hierarchy renewal	Send TPM2_ChangePPS command
TPM2_ChangeEPS	Zeroize the endorsement primary seed and flush all transient and persistent objects in the Endorsement hierarchy. Explicit zeroization indicator provided by service completion status.	Endorsement hierarchy renewal	Send TPM2_ChangeEPS command
TPM2_EvictControl	Zeroize an object from NVRAM. Explicit zeroization indicator provided by service completion status.	Method required to zeroize a dedicated object in NVRAM	Send TPM2_EvictControl command
TPM2_FlushContext	Zeroize an object from RAM. Explicit zeroization indicator provided by service completion status.	Method required to zeroize a dedicated object in RAM	Send TPM2_FlushContext command
Automatic	Zeroize SSPs at the end of a command processing. Implicit zeroization indication.	Method for limited life cycle SSPs	No, zeroization is automatic.
TPM2_NV_UndefineSpace TPM2_NV_UndefineSpaceSpecial	Zeroize a NV index. Explicit zeroization indicator provided by service completion status.	Method required to flush NV indices from NVRAM	Send TPM2_NV_UndefineSpace command. Send TPM2_NV_UndefineSpaceSpecial command
TPM2_VendorCmdZeroizeEK	Zeroize the endorsement key provisioned. Explicit zeroization indicator provided by service completion status.	Mandatory zeroization method for EK SSPs	Send TPM2_VendorCmdZeroizeEK command
TPM2_SequenceComplete TPM2_EventSequenceComplete	Zeroize a hash or HMAC sequence. Explicit zeroization indicator provided by service completion status.	Method required to flush sequences from RAM	Send TPM2_SequenceComplete command. Send TPM2_EventSequenceComplete command

Table 31: SSP Zeroization Methods

9.4 SSPs

All usage of these SSPs by the Module are described in the services detailed in section 4. Next table lists the SSPs used as keys:

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
contextEncKey	Wrapping key for context protection	256 - 256	Symmetric key - CSP	KBKDF		AES-ENC AES-DEC
contextKey	Derivation key for context protection	128 - 128	Symmetric key - CSP	DRBG		KBKDF
creHmacKey	HMAC key for credentials	160, 256, 384 - 128 to 256	Symmetric key - CSP	KBKDF		MAC KTS-Wrap KTS-Unwrap
creSeed	Seed for credential keys derivation. [CSP] Shared secret key that is established. [CSP] Key derivation function keys	160 to 384 - 128 to 256	Symmetric key - CSP		KAS	KBKDF
creSymKey	Encryption key for credentials	128, 192, 256 - 128 to 256	Symmetric key - CSP	KBKDF		AES-ENC AES-DEC KTS-

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
						Wrap KTS-Unwrap
drbgSeed	Entropy Input and Seed value for the DRBG	512 - 256	Seed - CSP	ENT-ESV		DRBG
drbgState	Internal state (V and C secret values) of the DRBG (based on SHA256)	256 - 256	State - CSP	DRBG		DRBG
dupInSymKey	Wrapping key for duplicated object	128, 192, 256 - 128 to 256	Symmetric key - CSP	DRBG		AES-ENC AES-DEC
dupOutHmacKey	HMAC key for duplicated objects	160, 256, 384 - 128 to 256	Symmetric key - CSP	KBKDF		MAC
dupOutSymKey	Encryption key for duplicated objects	128, 192, 256 - 128 to 256	Symmetric key - CSP	KBKDF		AES-ENC AES-DEC
dupSeed	Seed for protection keys derivation. [CSP] Shared secret key that is established. [CSP] Key derivation function keys	160 to 384 - 128 to 256	Symmetric key - CSP	DRBG KAS	KAS	KBKDF
eccParams256	[PSP] SP 800-56Arev3 domain parameters	256 - 128	Domain Parameters - PSP	Input during manufacturing		KAS
eccParams384	[PSP] SP 800-56Arev3 domain parameters	384 - 192	Domain Parameters - PSP	Input during manufacturing		KAS
ehProof	Proof (secret value) of the endorsement hierarchy	512 - 256	Symmetric key - CSP	DRBG		MAC
ehSeed	Seed of the endorsement hierarchy	512 - 256	Seed - CSP	ENT-ESV		DRBG
ekEcc	Provisioned ECC endorsement key. [CSP] Private key(s) in a key pair	256, 384 - 128 to 192	ECC private key - CSP	Input during manufacturing		KAS
ekRsa	Provisioned RSA endorsement key. [CSP] Private key(s) in a key pair	2048, 3072 - 112, 128	RSA private key - CSP	Input during manufacturing		KTS-Decap
endorsementAuth	Authentication value for the endorsement hierarchy	512 - 128 to 256 (depending on the underlying hash algorithm used)	Authentication value / Symmetric key - CSP			KBKDF MAC
ephPubEccKey	ECC ephemeral public key. [PSP] Public key(s) in a key pair	512, 768 - 128 to 192	ECC public key - PSP	KAS-KeyGen		KAS
ephSensEccKey	ECC ephemeral private key. [CSP] Private key(s) in a key pair	256, 384 - 128 to 192	ECC private key - CSP	KAS-KeyGen		KAS
extPubEccKey	ECC ephemeral public key. [PSP] Public key(s) in a key pair	512, 768 - 128 to 192	ECC public key - PSP			KAS
extSharedSecret	Output of KAS used externally to the TPM	256, 384 - 128, 192	Shared Secret - CSP		KAS	
fuSigECCKey	Field upgrade ECC signature verification key	384 - 192	ECC public key - PSP	Input during manufacturing		SigVer
fuSymKey	Non-SSP. Field upgrade symmetric key	256 - 256	Symmetric key - Neither	KBKDF		AES-DEC
fuSymSeed	Non-SSP. Seed used for field upgrade symmetric key derivation	256 - 256	N/A - Neither	Input during manufacturing		KBKDF
intSharedSecret	Output of KAS used internally to the TPM	256, 384 - 128, 192	Shared Secret - CSP		KAS	KAS
lockoutAuth	Authentication value for the lockout hierarchy	512 - 128 to 256 (depending on the underlying hash algorithm used)	Authentication value / Symmetric key - CSP			KBKDF MAC
nullProof	Proof (secret value) of the null hierarchy	512 - 256	Symmetric key - CSP	DRBG		KBKDF MAC
nullSeed	Seed of the null hierarchy	512 - 256	Seed - CSP	ENT-ESV		DRBG
nvAuth	Authorization of NV index	112 to 384 - 112 to 256	Authentication value / Symmetric key - CSP			KBKDF MAC
objAuth	Object's authorization value	112 to 384 - 112 to 256	Authentication value / Symmetric key - CSP			KBKDF MAC
objHmacKey	Integrity key of object private part	160, 256, 384 - 128 to 256	Symmetric key - CSP	KBKDF		MAC KTS-Wrap

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Size - Strength	Type - Category	Generated By	Established By	Used By
						KTS-Unwrap
objPub	Object public part. [PSP: Public key(s) in a key pair).	2048, 3072, 4096 (RSA) 2*256, 2*384 (ECC) - 112 to 256	Asymmetric public key - PSP	KeyGen KAS-KeyGen		SigVer KAS KTS-Encap
objSeed	Seed value for object generation	384 - 128 to 256	Data, Symmetric key - CSP	DRBG KBKDF		KBKDF SHA
objSens	Object private part. [CSP: Private key(s) in a key pair).	2048, 3072, 4096 (RSA) 128, 192, 256 (AES) 256, 384 (ECC) 112 to 1024 (HMAC) - 112 to 256	Symmetric or asymmetric private key - CSP	KeyGen KBKDF SKG KAS-KeyGen		AES-ENC AES-DEC SigGen KAS KTS-Decap KBKDF MAC
objSymKey	Encryption key of object private part	256 - 256	Symmetric key - CSP	KBKDF		AES-ENC AES-DEC KTS-Wrap KTS-Unwrap
ownerAuth	Authentication value for the storage hierarchy	512 - 128 to 256 (depending on the underlying hash algorithm used)	Authentication value / Symmetric key - CSP			KBKDF MAC
phProof	Proof (secret value) of the platform hierarchy	512 - 256	Symmetric key - CSP	DRBG		MAC
phSeed	Seed of the platform hierarchy	512 - 256	Seed - CSP	ENT-ESV		DRBG
platformAuth	Authentication value for the platform hierarchy	512 - 128 to 256 (depending on the underlying hash algorithm used)	Authentication value / Symmetric key - CSP			KBKDF MAC
seqAuth	Authorization value for hash or HMAC sequence	112 to 384 - 112 to 256	Authentication value / Symmetric key - CSP	N/A		KBKDF MAC
sesHmacKey	HMAC session key	160, 256, 384 - 128 to 256	Symmetric key - CSP	KBKDF		KBKDF MAC
sesSalt	Salt for keys diversification. [CSP] Shared secret key that is established. [CSP] Key derivation function keys	160, 256, 384 - 128 to 256	Symmetric key - CSP	N/A	KAS	KBKDF
sesSymKey	Encryption session key	128, 192, 256 - 128 to 256	Symmetric key - CSP	KBKDF		AES-ENC AES-DEC
shProof	Proof (secret value) of the storage hierarchy	512 - 256	Symmetric key - CSP	DRBG		KBKDF MAC
shProofForReseed	Random value	512 - 256	Entropy source - CSP	ENT-ESV		DRBG
shSeed	Seed of the storage hierarchy	512 - 256	Seed - CSP	ENT-ESV		DRBG
tdrbgState	Internal state (V and C secret values) of the transient DRBG (based on SHA256) used to generate prime numbers for primary RSA keys	256 - 256	State - CSP	DRBG		DRBG

Table 32: SSP Table 1

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
contextEncKey		Dynamic RAM:Plaintext	After Use	Automatic	contextKey:Derives nullProof:Derives phProof:Derives ehProof:Derives shProof:Derives
contextKey		Static RAM:Plaintext	Until next reset	TPM2_Init	drbgState:Generates contextEncKey:Derived From
creHmacKey		Dynamic RAM:Plaintext	After Use	Automatic	creSeed:Derived From
creSeed	Input asym. encrypted to RAM Output asym. encrypted from RAM	Dynamic RAM:Plaintext	After Use	Automatic	creSymKey:Derives creHmacKey:Derives objPub:Encrypts intSharedSecret:Derives
creSymKey		Dynamic RAM:Plaintext	After Use	Automatic	creSeed:Derived From
drbgSeed		Dynamic RAM:Plaintext	After Use	Automatic	drbgState:Instantiated with
drbgState		Static RAM:Plaintext	Until next reset or use of TPM2_Clear	TPM2_Init TPM2_Clear	drbgSeed:Instantiates
dupInSymKey	Input plaintext to RAM Input protected to RAM Output plaintext from RAM Output protected from RAM	Dynamic RAM:Plaintext	After Use	Automatic	sesSymKey:Encrypts sesHmacKey:Protects (Integrity) objSens:Encrypted by
dupOutHmacKey		Dynamic RAM:Plaintext	After Use	Automatic	dupSeed:Derives objSens:Protects (Integrity) objAuth:Protects (Integrity) objSeed:Protects (Integrity)
dupOutSymKey		Dynamic RAM:Plaintext	After Use	Automatic	dupSeed:Derives objSens:Encrypted by objAuth:Encrypted by objSeed:Encrypted by
dupSeed	Input asym. encrypted to RAM Output asym. encrypted from RAM	Dynamic RAM:Plaintext	After Use	Automatic	objPub:Encrypts dupOutSymKey:Derived from dupOutHmacKey:Derived from intSharedSecret:Derives
eccParams256		NVRAM:Plaintext	After Use	N/A	ekEcc:Derived From ephPubEccKey:Derived From ephSensEccKey:Derived From objPub:Derived From objSens:Derived From
eccParams384		NVRAM:Plaintext	After Use	N/A	ekEcc:Derived From ephPubEccKey:Derived From ephSensEccKey:Derived From objPub:Derived From objSens:Derived From

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
ehProof		NVRAM:Plaintext	After Use	TPM2_ChangeEPS TPM2_Clear	drbgState:Generates contextEncKey:Derived From
ehSeed		NVRAM:Plaintext	After Use	TPM2_ChangeEPS	tdrbgState:Instantiated with
ekEcc		NVRAM:Plaintext	After Use	TPM2_VendorCmdZeroizeEK	objSens:Derived From
ekRsa		NVRAM:Plaintext	After Use	TPM2_VendorCmdZeroizeEK	objSens:Derived From
endorsementAuth	Input plaintext to NVRAM Input protected to NVRAM	NVRAM:Plaintext	After Use	TPM2_Clear TPM2_ChangeEPS	sesHmacKey:Derived from, Protects (Integrity) sesSymKey:Derived from, Protects (Encrypts)
ephPubEccKey	Input plaintext to RAM Output plaintext from RAM	Dynamic RAM:Plaintext	After Use	Automatic	ephSensEccKey:Derives extSharedSecret:Derives intSharedSecret:Derives
ephSensEccKey		Dynamic RAM:Plaintext	After Use	Automatic	drbgState:Generates extSharedSecret:Derives
extPubEccKey	Input plaintext to RAM	Dynamic RAM:Plaintext	After Use	Automatic	sesSalt:Derived from
extSharedSecret	Input plaintext to RAM Input protected to RAM Output plaintext from RAM Output protected from RAM	Dynamic RAM:Plaintext	After Use	Automatic	ephPubEccKey:Derived From ephSensEccKey:Derived From objPub:Derived From objSens:Derived From
fuSigECCKey		NVRAM:Plaintext	After Use	N/A	
fuSymKey		Dynamic RAM:Plaintext	After Use	Automatic	fuSymSeed:Derives
fuSymSeed		NVRAM:Plaintext	After Use	N/A	fuSymKey:Derived From
intSharedSecret		Dynamic RAM:Plaintext	After Use	Automatic	extPubEccKey:Derived From objSens:Derived From
lockoutAuth	Input plaintext to NVRAM Input protected to NVRAM	NVRAM:Plaintext	After Use	TPM2_Clear	sesHmacKey:Derived from, Protects (Integrity) sesSymKey:Derived from, Protects (Encrypts)
nullProof		Static RAM:Plaintext	Until next reset	TPM2_Init	drbgState:Generates contextEncKey:Derived From
nullSeed		Static RAM:Plaintext	Until next reset	TPM2_Init	tdrbgState:Instantiated with
nvAuth	Input plaintext to NVRAM Input protected to NVRAM	NVRAM:Plaintext	After Use	TPM2_NV_UndefineSpace TPM2_NV_UndefineSpaceSpecial	sesHmacKey:Derived from, Protects (Integrity) sesSymKey:Encrypts
objAuth	Input plaintext to RAM Input protected to RAM Output protected from RAM Output protected from NVRAM	Static RAM:Plaintext NVRAM:Plaintext	Until object zeroization, shift to NVRAM or next reset	TPM2_Init TPM2_Clear TPM2_ChangePPS TPM2_ChangeEPS TPM2_EvictControl TPM2_FlushContext	sesHmacKey:Derived from, Protects (Integrity) sesSymKey:Derived from, Encrypts
objHmacKey		Dynamic RAM:Encrypted NVRAM:Plaintext	After Use	Automatic	objAuth:Protected by (Integrity) objSens:Protected by (Integrity) platformAuth:Protected by (Integrity)

Public Material – May be reproduced only in its original entirety (without revision).

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					endorsementAuth:Protected by (Integrity) ownerAuth:Protected by (Integrity) lockoutAuth:Protected by (Integrity)
objPub	Input plaintext to RAM Output plaintext from NVRAM Output plaintext from RAM	Static RAM:Plaintext NVRAM:Plaintext	Until object zeroization, shift to NVRAM or next reset	TPM2_Init TPM2_Clear TPM2_ChangePPS TPM2_ChangeEPS TPM2_EvictControl TPM2_FlushContext	objSens:Paired With extSharedSecret:Derives
objSeed	Input protected to RAM Input plaintext to RAM Output protected from RAM Output protected from NVRAM	Static RAM:Plaintext NVRAM:Plaintext	Until object zeroization, shift to NVRAM or next reset	TPM2_Init TPM2_Clear TPM2_ChangePPS TPM2_ChangeEPS TPM2_EvictControl TPM2_FlushContext	tdrbgState:Derived From drbgState:Derived From objSymKey:Derived From objHmacKey:Derived From sesHmacKey:Protects (Integrity) sesSymKey:Protects (Encrypts)
objSens	Input plaintext to RAM Input protected to RAM Output protected from RAM Output protected from NVRAM	Static RAM:Plaintext NVRAM:Plaintext	Until object zeroization, shift to NVRAM or next reset	TPM2_Init TPM2_Clear TPM2_ChangePPS TPM2_ChangeEPS TPM2_EvictControl TPM2_FlushContext	tdrbgState:Generates drbgState:Generates objSeed:Derives objSymKey:Encrypts objHmacKey:Protects (Integrity) objPub:Paired With extSharedSecret:Derives intSharedSecret:Derives
objSymKey		Dynamic RAM:Plaintext NVRAM:Plaintext	After Use	Automatic	objAuth:Encrypted by objSens:Encrypted by platformAuth:Encrypted by endorsementAuth:Encrypted by ownerAuth:Encrypted by lockoutAuth:Encrypted by
ownerAuth	Input plaintext to NVRAM Input protected to NVRAM	NVRAM:Plaintext	After Use	TPM2_Clear	sesHmacKey:Derived from, Protects (Integrity) sesSymKey:Derived from, Protects (Encrypts)
phProof		NVRAM:Plaintext	After Use	TPM2_ChangePPS	drbgState:Generates contextEncKey:Derived From
phSeed		NVRAM:Plaintext	After Use	TPM2_ChangePPS	tdrbgState:Instantiated with
platformAuth	Input plaintext to RAM Input protected to RAM	Static RAM:Plaintext	Until next reset	TPM2_Init	sesHmacKey:Derived from, Protects (Integrity) sesSymKey:Encrypts
seqAuth	Input plaintext to RAM Input protected to RAM Output protected from RAM	Dynamic RAM:Plaintext	Until use of zeroization command or next reset	TPM2_SequenceComplete TPM2_EventSequenceComplete	sesSymKey:Derived From sesHmacKey:Derived From
sesHmacKey	Input protected to RAM Output protected from RAM	Dynamic RAM:Plaintext	After Use	Automatic	nvAuth:Protected by (Integrity) contextKey:Encrypts contextEncKey:Encrypts platformAuth:Protected by (Integrity) endorsementAuth:Protected by

Public Material – May be reproduced only in its original entirety (without revision).

Name	Input - Output	Storage	Storage Duration	Zeroization	Related SSPs
					(Integrity) ownerAuth:Protected by (Integrity) lockoutAuth:Protected by (Integrity) objAuth:Protected by (Integrity) sesSalt:Derives seqAuth:Derives; Protected by (Integrity) dupInSymKey:Protected by (Integrity)
sesSalt	Input asym. encrypted to RAM	Dynamic RAM:Plaintext	After Use	Automatic	sesHmacKey:Derived From objPub:Encrypts intSharedSecret:Derives
sesSymKey		Dynamic RAM:Plaintext	After Use	Automatic	sesHmacKey:Derives platformAuth:Derives; Encrypts endorsementAuth:Derives; Encrypts ownerAuth:Derives; Encrypts lockoutAuth:Derives; Encrypts objAuth:Derives; Encrypts seqAuth:Derives; Encrypts nvAuth:Derives; Encrypts dupInSymKey:Encrypted by
shProof		NVRAM:Plaintext	After Use	TPM2_Clear	drbgState:Generates contextEncKey:Derived From
shProofForReseed		NVRAM:Plaintext	After Use	TPM2_Clear	tdrbgState:Reseeded From
shSeed		NVRAM:Plaintext	After Use	TPM2_Clear	tdrbgState:Instantiated with
tdrbgState		Dynamic RAM:Plaintext	After Use	Automatic	nullSeed:Instantiates phSeed:Instantiates ehSeed:Instantiates shSeed:Instantiates

Table 33: SSP Table 2

Next table gives the security strength of a key depending on the underlying algorithm used and its size:

Algorithm*	Underlying algorithm	Key size (bits)	Security strength (bits)
KBKDF	SHA-1	size ≥ 128	128
		size < 128	Key size
	SHA2-256	size ≥ 192	192
		size < 192	Key size
	SHA2-384	size ≥ 256	256
		size < 256	Key size
HMAC	SHA-1	size ≥ 128	128
		size < 128	Key size
	SHA2-256	size ≥ 192	192
		size < 192	Key size
	SHA2-384	size ≥ 256	256
		size < 256	Key size
DRBG	SHA2-256	-	256
AES	-	128 / 192 / 256	128 / 192 / 256
RSA	-	2048 / 3072 / 4096	112 / 128 / 142
ECC	-	256 / 384	128 / 192

Table 34 – Security Strength of a Key Depending on the Underlying Algorithm Used and its Size

9.5 Transitions

SHA-1: The module only supports the use of SHA-1 in the Approved mode of operation for non-digital signature applications as permitted by NIST SP 800-131Ar2. The module only permits the use of SHA-1 for digital signatures in the non-Approved mode. Per CMVP Programmatic Transitions and NIST SP 800-131Ar2, usage of SHA-1 for non-digital-signature applications is deprecated through Dec 31, 2030.

Minimum key size transition: In accordance with NIST SP 800-131Ar2, the module meets the minimum key size requirement of 112. NIST SP 800-57, Part 1 includes a transition to a security strength of 128 bits from 2031 and beyond.

9.6 Additional Information

N/A

Public Material – May be reproduced only in its original entirety (without revision).

10 Self-Tests

10.1 Pre-Operational Self-Tests

The Module performs self-tests to ensure the proper operation of the Module. Per FIPS 140-3 these are categorized as either pre-operational self-tests or conditional self-tests. Pre-operational self–tests are available on demand by power cycling the Module. The Module performs the following pre-operational self-tests in the table below:

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details
Firmware integrity test	CRC 16	EDC	SW/FW Integrity	Successful execution of TPM2_Startup command indicates tests have been run	FW integrity is verified by computing an EDC (CRC-16 [ISO13239]) and comparing it to reference values.
HW integrity	HW registers verification	KAT	Critical Function	Successful execution of TPM2_Startup command indicates tests have been run	HW integrity is guaranteed via check of HW sensors. If failure is detected during boot sequence, status is set to FAIL, and error is returned.

Table 35: Pre-Operational Self-Tests

10.2 Conditional Self-Tests

The Module performs the following conditional self-tests in the table below. The bit index indicated in the “Indicator” column corresponds to the index in the algos_status field in the TPM2_GetTestResult response.

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
AES-DEC (A2553)	AES-128-CBC	KAT	CAST	Bit #7 clear	AES CBC 128 decryption of known encrypted data and comparison to the expected plaintext data	Power On
AES-ENC (A2553)	AES-128-CBC	KAT	CAST	Bit #7 clear	AES CBC 128 encryption of known data compared to a reference value.	Power On
ECDSA KeyGen (FIPS186-4) (A2555)	P-256, P-384	PCT	PCT	Key creation failure	Depending on the key purpose (signing or key establishment) an ECDSA signature is generated (k fixed and the message varies) and verified with pairwise consistency test as defined by [56Ar3] or a scalar multiplication is done and compared to the public key.	Upon ECC Key Generation
ECDSA SigGen (FIPS186-4) (A2555)	NIST P-256	KAT	CAST	Bit #10 clear	ECDSA signature generation on known data with known key and k. Output of signature is compared to a reference signature.	Power On
ECDSA SigVer (FIPS186-4) (A2555)	NIST P-256	KAT	CAST	Bit #10 clear	ECDSA signature verification on known signature with known key and k.	Power On
Entropy	RCT and APT	[90B] Health-Tests	CAST	Bit #1 clear	AIS31 and [90B] (RCT and APT) start-up health tests on ESV #E41 output sequence. If test fails, test status is set to FAIL, and error is returned	At each random bits generation
Firmware loading	ECDSA P-384	Signature Verification	SW/FW Load	Error returned on FW loading command	Verification of chained digest and signature to ensure authentication of the FW	Upon firmware load
HMAC-SHA-1 (A2551, A552)	HMAC-SHA1	KAT	CAST	Bit #5 clear	HMAC on known data and known key. Comparison of output to an expected MAC value (20 bytes)	Power On
Hash DRBG (A2547)	SHA2-256	KAT	CAST	Bit #1 clear	Instantiate then Reseed are seeded with a known seed value (64 bytes). Random is then generated with Generate API to output a 32-bytes value compared to a reference value (single test sequence done in accordance with Section 11.3 of [90A])	Power On
KAS-ECC Sp800-56Ar3 (A2555)	NIST P-256	KAT	CAST	Bit #9 clear	Primitive "Z" Computation and key derivation are implemented: a known private key d is used with a known point P of NIST P-256 curve to compute Q = dP. Key derivation of Q performed with SHA-1 underlying algorithm to output a key of 20 bytes that is compared to a refence value	Power On

Algorithm or Test	Test Properties	Test Method	Test Type	Indicator	Details	Conditions
KDF SP800-108 (A2550)	N/A	KAT	CAST	Bit #6 clear	KDF on known data and known label. Comparison of output to an expected derivation value (32 bytes)	Power On
RSA KeyGen (FIPS186-5) (A2554)	2048, 3072 or 4096-bit	PCT	PCT	Key creation failure	Depending on the key purpose (signing or encrypting) indicated in sign attribute of the key, encryption/decryption or signing/verification is done on known data	Upon RSA Key Generation
RSA SigGen (FIPS186-5) (A2554)	RSASSA-PKCS1-v1_5	KAT	CAST	Bit #12 clear	RSA signature generation on known data with a known key. Output of signature is compared to a reference signature (covers also KTS-IFC functionality)	Power On
RSA SigVer (FIPS186-5) (A2554)	RSASSA-PKCS1-v1_5	KAT	CAST	Bit #12 clear	RSA signature verification on a known signature with a known key (covers also KTS-IFC functionality)	Power On
SHS (A2548, A2549)	SHA1, SHA2-256, SHA2-384, SHA3-256	KAT	CAST	Bit #2 clear Bit #3 clear Bit #4 clear	Hash of known data and comparison of output to an expected digest. SHA-1, SHA2-256, SHA2-384 are tested twice to cover each of the two implementations covered by CAVP Cert. #A2548 and #A2549.	Power On

Table 36: Conditional Self-Tests

Public Material – May be reproduced only in its original entirety (without revision).

10.3 Periodic Self-Test Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
Firmware integrity test	EDC	SW/FW Integrity	On demand	Manually
HW integrity	KAT	Critical Function	On demand	Manually

Table 37: Pre-Operational Periodic Information

Algorithm or Test	Test Method	Test Type	Period	Periodic Method
AES-DEC (A2553)	KAT	CAST	On Demand	Manually
AES-ENC (A2553)	KAT	CAST	On Demand	Manually
ECDSA KeyGen (FIPS186-4) (A2555)	PCT	PCT	N/A	Manually
ECDSA SigGen (FIPS186-4) (A2555)	KAT	CAST	On Demand	Manually
ECDSA SigVer (FIPS186-4) (A2555)	KAT	CAST	On Demand	Manually
Entropy	[90B] Health-Tests	CAST	On Demand	Manually
Firmware loading	Signature Verification	SW/FW Load	On Demand	Manually
HMAC-SHA-1 (A2551, A552)	KAT	CAST	On Demand	Manually
Hash DRBG (A2547)	KAT	CAST	On Demand	Manually
KAS-ECC Sp800-56Ar3 (A2555)	KAT	CAST	On Demand	Manually
KDF SP800-108 (A2550)	KAT	CAST	On Demand	Manually
RSA KeyGen (FIPS186-5) (A2554)	PCT	PCT	N/A	Manually
RSA SigGen (FIPS186-5) (A2554)	KAT	CAST	On Demand	Manually
RSA SigVer (FIPS186-5) (A2554)	KAT	CAST	On Demand	Manually
SHS (A2548, A2549)	KAT	CAST	On Demand	Manually

Table 38: Conditional Periodic Information

10.4 Error States

Public Material – May be reproduced only in its original entirety (without revision).

Name	Description	Conditions	Recovery Method	Indicator
ES1	The Module fails a KAT, PCT, FW or HW integrity verification, [90B] health test	The Module enters the failure state	Reboot/Power cycle the module	Outputs return code of TPM_RC_FAILURE, otherwise it indicates successful completion by TPM_RC_SUCCESS
ES2	The Module fails a firmware loading test	The Module returns to normal state	None	Return code different from TPM_RC_SUCCESS sent on firmware upgrade start command

Table 39: Error States

All cryptographic functions are inhibited while the module is in an error state. Successful completion of self-tests can be verified through use of TPM2_GetTestResult command. The first 4 bytes of response indicate self-tests status. If they are equal to 0, self-tests completed successfully. If not, the subsequent 4 bytes indicate the list of algorithms not fully self-tested.

Public Material – May be reproduced only in its original entirety (without revision).

11 Life-Cycle Assurance

11.1 Installation, Initialization, and Startup Procedures

Installation and Initialization:

The following steps must be performed in order to securely install, initialize, and start up the Module in FIPS 140-3 Approved mode of operation:

- Connection of the Module with its environment must be done according to the pinout description detailed in section 3.2

Module operation:

TPM is operated in an approved mode of operation as long as no non-approved service using a non-approved algorithm is used. No specific rules of operation are required to operate this module at FIPS 140-3 Security Level 1.

TPM is in normal operation mode when all pre-operational and conditional self-tests (apart from firmware load and PCT tests) are complete. All approved and non-approved services with the corresponding indicator reporting if the service uses an approved cryptographic algorithm or a security function.

11.2 Administrator Guidance

No specific initialization procedure is required.

11.3 Non-Administrator Guidance

No initialization procedures are required.

11.4 Design and Rules

Rules of Operation

1. The Module provides two operator roles: the Cryptographic Officer and the User role. Each role is associated with a set of services as detailed in Section 4.3.
2. The Module requires authentication to access some of the services as detailed in Section 4.1, but this is not assessed as part of this FIPS 140-3 Security Level 1 evaluation.
3. The Module allows the operator to initiate power-up self-tests by power cycling or resetting the Module.
4. Power up self-tests do not require any operator action.
5. Data output is inhibited during key generation, self-tests, zeroisation, firmware loading, and error states.
6. Status information does not contain CSPs or sensitive data that if misused could lead to a compromise of the Module.

Public Material – May be reproduced only in its original entirety (without revision).

7. The Module does not support concurrent operators.
8. The Module does not support a maintenance interface or role.
9. The Module does not support manual key entry method.
10. The Module does not have any proprietary external input/output devices used for entry/output of data.
11. The Module does not output intermediate key values.
12. The Module does not provide bypass services or ports/interfaces.
13. The Module does not support a self-initiated cryptographic output capability.
14. For all zeroisation methods, the module must be in direct control of the operator.

11.5 Maintenance Requirements

N/A

11.6 End of Life

End-of-life of the product requires the following zeroisation commands to be executed to remove all CSPs from the memory of the module:

- TPM2_Init
- TPM2_Clear
- TPM2_ChangeEPS
- TPM2_ChangePPS
- TPM2_VendorCmdZeroizeEK
- TPM2_NV_UndefineSpace or TPM2_NV_UndefineSpaceSpecial

Public Material – May be reproduced only in its original entirety (without revision).

12 Mitigation of Other Attacks

The Module does not implement any mitigation method against other attacks.

Public Material – May be reproduced only in its original entirety (without revision).

References and Definitions

The following standards are referred to in this Security Policy.

Abbreviation*	Full Specification Name
[TPM2.0 Part1]	*TPM2.0 Main, Part 1, Architecture, rev 1.59, TCG*
[TPM2.0 Part2]	*TPM2.0 Main, Part 2, Structures, rev 1.59, TCG*
[TPM2.0 Part3]	*TPM2.0 Main, Part 3, Commands, rev 1.59, TCG*
[TPM2.0 Part4]	*TPM2.0 Main, Part 4, Supporting routines, rev 1.59, TCG*
[PTP 1.06]	*TCG PC Client Platform TPM Profile (PTP) Specification, rev. 1.06*
[FIPS TCG]	*TCG FIPS 140-3 guidance for TPM 2.0, version 1.0, rev. 1, January 30, 2024*
[ISO19790]	*International Standard, ISO/IEC 19790, Information technology — Security techniques — Test requirements for cryptographic modules, Third edition, March 2017*
[ISO24759]	*International Standard, ISO/IEC 24759, Information technology — Security techniques — Test requirements for cryptographic modules, Second and Corrected version, 15 December 2015*
[ISO13239]	*International Standard, ISO/IEC 13239, Information technology — Telecommunications and information exchange between systems — High-level data link control (HDLC) procedures, July 2002*
[140-3]	*Security Requirements for Cryptographic Modules, March 22, 2019*
[140]	*NIST Special Publication 800-140, FIPS 140-3 Derived Test Requirements (DTR), CMVP Validation Authority Updates to ISO/IEC 24759, March 2020*
[140A]	*NIST Special Publication 800-140A, CMVP Documentation Requirements, CMVP Validation Authority Updates to ISO/IEC 24759, March 2020*
[140Br1]	*NIST Special Publication 800-140B revision 1, CMVP Security Policy Requirements, CMVP Validation Authority Updates to ISO/IEC 24759 and ISO/IEC 19790 Annex B, November 2023*
[140C]	*NIST Special Publication 800-140Cr2, CMVP Approved Security Functions, CMVP Validation Authority Updates to ISO/IEC 24759, July 2023*
[140D]	*NIST Special Publication 800-140Dr2, CMVP Approved Sensitive Security Parameter Generation and Establishment Methods, CMVP Validation Authority Updates to ISO/IEC 24759, July 2023*
[140E]	*NIST Special Publication 800-140E, CMVP Approved Authentication Mechanisms, CMVP Validation Authority Requirements for ISO/IEC 19790:2012 Annex E and ISO/IEC 24759 Section 6.17, March 2020*
[140F]	*NIST Special Publication 800-140Fr1, CMVP Approved Non-Invasive Attack Mitigation Test Metrics, CMVP Validation Authority Updates to ISO/IEC 24759, August 2021*
[IG]	*Implementation Guidance for FIPS PUB 140-3 and the Cryptographic Module Validation Program, October 23, 2024*
[108]	*NIST Special Publication 800-108r1-upd1, Recommendation for Key Derivation Using Pseudorandom Functions (Revised), August 2022*
[131A]	*Transitions: Recommendation for Transitioning the Use of Cryptographic Algorithms and Key Lengths, Revision 2, March 2019*

Public Material – May be reproduced only in its original entirety (without revision).

Abbreviation*	Full Specification Name
[133]	*NIST Special Publication 800-133, Recommendation for Cryptographic Key Generation, Revision 2, June 2020*
[135]	*National Institute of Standards and Technology, Recommendation for Existing Application-Specific Key Derivation Functions, Special Publication 800-135rev1, December 2011*
[186]	*National Institute of Standards and Technology, Digital Signature Standard (DSS), Federal Information Processing Standards Publication 186-5, Feb 2023*
[197]	*National Institute of Standards and Technology, Advanced Encryption Standard (AES), Federal Information Processing Standards Publication 197-upd1, May, 2023*
[198]	*National Institute of Standards and Technology, The Keyed-Hash Message Authentication Code (HMAC), Federal Information Processing Standards Publication 198-1, July, 2008*
[180]	*National Institute of Standards and Technology, Secure Hash Standard, Federal Information Processing Standards Publication 180-4, August 2015*
[202]	*FEDERAL INFORMATION PROCESSING STANDARDS PUBLICATION, SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions, FIPS PUB 202, August 2015*
[208]	*National Institute of Standards and Technology, Recommendation for Stateful Hash-Based Signature Schemes, October 2020*
[38A]	*National Institute of Standards and Technology, Recommendation for Block Cipher Modes of Operation, Methods and Techniques, Special Publication 800-38A, December 2001*
[56Ar3]	*NIST Special Publication 800-56A Revision 3, Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography, April 2018*
[56Br2]	*NIST Special Publication 800-56B Revision 2, Recommendation for Pair-Wise Key Establishment Schemes Using Finite Field Cryptography, March 2019*
[90A]	*National Institute of Standards and Technology, Recommendation for Random Number Generation Using Deterministic Random Bit Generators, Special Publication 800-90A, Revision 1, June 2015*
[90B]	*National Institute of Standards and Technology, Recommendation for the Entropy Sources Used for Random Bit Generation, Special Publication 800-90B, January 2018*

Table 40 – References

Acronym*	Definition
APT	Adaptive Proportion Test
BN P-256	Barreto-Naehrig 256-bit elliptic curve
FW	Firmware
HW	Hardware
KAT	Know Answer Test
I^2C	Inter-Integrated Circuit
MPU	Memory Protection Unit
RCT	Repetition Count Test
SPI	Serial Peripheral Interface

Public Material – May be reproduced only in its original entirety (without revision).

Acronym*	Definition
SSP	Sensitive Security Parameter
TCG	Trusted Computing Group
TPM	Trusted Platform Module

Table 41 – Acronyms and Definitions

Public Material – May be reproduced only in its original entirety (without revision).